

HKOI Lesson VIII – 18th July 2006
Answer to previous lesson

Answers for HKOI'99 Heat Event (Junior Group)

1999hje.doc

1	B	11	D	21	A	31	A	41	B
2	E	12	E	22	A	32	C	42	B
3	E	13	D	23	E	33	D	43	C
4	B	14	A	24	B	34	A	44	E
5	B	15	C	25	B	35	C	45	A

6	C	16	E	26	C	36	C
7	A	17	B	27	D	37	C
8	D	18	D	28	C	38	B
9	B	19	C	29	A	39	D
10	E	20	E	30	A	40	D

Answer for Lesson3: G1&2Quiz.doc

1)

Line	Correction
3	Var a,b:integer;
4	C:string;
7	Writeln(c);
8	For a:=1 to 10 do
11	Write(a*b);
13	End;
14	End.

Notes: All things after end. (full-stop) will not be considered , so no need to correct line 15.

Besides, semi-colon is not necessary just before the reserved word end, so line 12 has no error.

Also, in the sixth line, that pair of brackets will not cause any syntax error.

You may think of the statement "a:=(9);", which is actually a valid statement that assigns 9 to a.

2)

```

Program enumeration;
Uses winCRT;
Var I,j,n,t:integer;
Begin
readln(n);
for I:=1 to n do
begin
t:=i; write(t);
for j:=1 to I-1 do
begin
t:=t*I;
write(' ',t);
end;
writeln;
end;
End.
    
```

3)

```

Program Fibonacci;
Uses winCRT;
Var a,b,I,n:integer;
Begin
readln(n);
a:=1;
b:=1;
write(a, ' ',b);
for I:=3 to n do
begin
a:=a+b;
b:=a-b;
write(' ',a);
end;
writeln;
End.
    
```

Notes: Since there is no any extra space before end of line, so we have to use the above method to provide the right format.

Answer for Lesson4.doc:

Exercise III: Lightout

```
var map:array[0..11,0..11] of boolean;
    m,n,x,y,t,i,j:integer;
begin
  readln(n);
  fillchar(map,sizeof(map),0);
  for i:=1 to n do
  begin
    read(j);
    for j:=1 to j do
    begin
      read(t);
      map[i,t]:=true;
    end;
  end;
  readln(m);
  {Continue on R.H.S.}

  {fillchar(map,sizeof(map),0) means
  store 0 to each element of map}
```

```
for i:=1 to m do {Continue from L.H.S.}
begin
  readln(x,y);
  map[x,y]:=not map[x,y];
  map[x+1,y]:=not map[x+1,y];
  map[x-1,y]:=not map[x-1,y];
  map[x,y+1]:=not map[x,y+1];
  map[x,y-1]:=not map[x,y-1];
end;
for i:=1 to n do
begin
  t:=0;
  for j:=1 to n do t:=t+byte(map[i,j]);
  write(t);
  for j:=1 to n do if map[i,j] then write(' ');
  writeln;
end;
end.
```

Notes: The array [0..11,0..11] is to prevent out-of-range error, i.e. to simplify the boundary case
So, we don't need to check for $x > 1$ to execute the code $\text{map}[x-1,y] := \text{not map}[x-1,y]$ etc.

This question is quoted from HKOI2003 Practice Problem

This is an official sample solution

[<http://www.hkoi.org/2003/lightout1.html>]

Exercise IV: Bridge

```
type rec=record
  x1,y1,x2,y2:integer;
end;
var fi,fo:text;
    a,b:rec;
    t:array[0..4] of real;
    i:integer;
    x,y:boolean;
begin
  assign(fi,'INPUT4.txt');reset(fi);
  assign(fo,'OUTPUT4.txt');rewrite(fo);
  readln(fi,a.x1,a.y1,a.x2,a.y2);
  readln(fi,b.x1,b.y1,b.x2,b.y2);
  x:=false; y:=false;
  for i:=a.x1 to a.x2 do
  if i in [b.x1..b.x2] then x:=true;
  for i:=a.y1 to a.y2 do
  if i in [b.y1..b.y2] then y:=true;

  {Continue on R.H.S.}
```

```
if x then {Continue from L.H.S.}
begin
  t[1]:=abs(a.y2-b.y1);
  t[2]:=abs(b.y2-a.y1);
  writeln(fo,(t[1]+t[2]-abs(t[1]-t[2]))/2:0:3);
end else if y then
begin
  t[1]:=abs(a.x2-b.x1);
  t[2]:=abs(b.x2-a.x1);
  writeln(fo,(t[1]+t[2]-abs(t[1]-t[2]))/2:0:3);
end else
begin
  t[1]:=sqrt(sqr(a.x2-b.x1)+sqr(a.y2-b.y1));
  t[2]:=sqrt(sqr(b.x2-a.x1)+sqr(b.y2-a.y1));
  t[3]:=sqrt(sqr(a.x1-b.x2)+sqr(a.y2-b.y1));
  t[4]:=sqrt(sqr(b.x1-a.x2)+sqr(b.y2-a.y1));
  t[0]:=9.9E+10;
  for i:=1 to 4 do
  {if t[i]<t[0] then t[0]:=t[i];}
  writeln(fo,t[0]:0:3);
end;
close(fi);close(fo);
end.
```

Notes: The first line can be the coordinate of the top-right rectangle.

Also, a straight line must be the shortest path between them, when they are side-by-side etc.

Answer for Lesson5.doc : String.doc

4bi

```
var s:string;
    i:integer;
begin
  readln(s);
  for i:=1 to length(s) do
  begin
    writeln(copy(s,1,i));
    s[i]:=' ';
  end;
end.
```

4bii

```
var s:string;
    i:integer;
begin
  readln(s);
  for i:=1 to length(s) do
  write(s[length(s)-i+1]);
  writeln;
end.
```

5

```
var s:string;
    c1,c2,i,t:integer;
begin
  writeln('Enter a sentence:');
  readln(s);
  writeln('*** Sentence Statistics ***');
  t:=pos(' ',s); c1:=1;
  while t<>0 do
  begin
    delete(s,t,1);
    inc(c1);
    t:=pos(' ',s);
  end;
  c2:=0;
  for i:=1 to length(s) do
  if upcase(s[i]) in ['A'..'Z'] then inc(c2);
  writeln('Number of words: ',c1);
  writeln('Number of letters: ',c2);
end.
```

6

```
var s:string;
    x,e:integer;
begin
  repeat
  write('Please enter an integer: ');
  readln(s);
  val(s,x,e);
  if e<>0 then writeln('Error! Invalid input. ');
  until e=0;
  writeln('Correct!');
end.
```

7.

```
var s:string;
    i,j:integer;
begin
  readln(s);
  for i:=1 to length(s) do
  begin
    for j:=1 to length(s)-i do write(#32);
    writeln(copy(s,length(s)-
    i+1,i)+copy(s,1,i));
  end;
  for i:=1 to length(s) div 2 do
  begin
    for j:=1 to length(s)-1 do write(#32);
    writeln(s[length(s)]+s[1]);
  end;
end.
```

8.

```
var s1,s2:string;
    x:integer;
begin
  write('Please enter a sentence: ');
  readln(s1);
  write('Please enter a word: ');
  readln(s2);
  write('Please enter an insertion position: ');
  readln(x);
  s1:=copy(s1,1,x-1)+s2
  +copy(s1,x+1,length(s1)-x);
  writeln('The new sentence is : ',s1);
end.
```

Answer for Lesson6_1.doc: Practical Exercise:

1

```
var ans,s:string; i:integer;
begin
  readln(s); ans:="";
  for i:=1 to length(s) div 2 do
    if copy(s,1,i)=copy(s,i+1,i) then ans:=copy(s,1,i);
  if ans <> "" then writeln(ans)
  else writeln('No Repeating Prefix.');
```

3

```
var cnt:array[#0..#255] of integer;
    s:string; i:integer; c:char;
begin
  fillchar(cnt,sizeof(cnt),0);
  readln(s);
  for i:=1 to length(s) do inc(cnt[uppercase(s[i])]);
  for c:='A' to 'Z' do if cnt[c] > 0 then writeln(c,' ',cnt[c]);
end.
```

2

```
var rom:array[0..3,0..9] of string[5];
    t,n:integer;
    s:string;
begin
  fillchar(rom,sizeof(rom),0);
  rom[0,1]:='I'; rom[1,1]:='X'; rom[2,1]:='C'; rom[3,1]:='M';
  rom[0,2]:='II'; rom[1,2]:='XX'; rom[2,2]:='CC'; rom[3,2]:='MM';
  rom[0,3]:='III'; rom[1,3]:='XXX'; rom[2,3]:='CCC'; rom[3,3]:='MMM';
  rom[0,4]:='IV'; rom[1,4]:='XL'; rom[2,4]:='CD';
  rom[0,5]:='V'; rom[1,5]:='L'; rom[2,5]:='D';
  rom[0,6]:='VI'; rom[1,6]:='LX'; rom[2,6]:='DC';
  rom[0,7]:='VII'; rom[1,7]:='LXX'; rom[2,7]:='DCC';
  rom[0,8]:='VIII'; rom[1,8]:='LXXX'; rom[2,8]:='DCCC';
  rom[0,9]:='IX'; rom[1,9]:='XC'; rom[2,9]:='CM';
  s:="";
  t:=0;
  readln(n);
  while n <> 0 do
  begin
    s:=rom[t,n mod 10]+s;
    n:=n div 10;
    inc(t);
  end;
  writeln(s);
end.
```

4) There are only two pair of integers fulfill the condition (2,3,17) , (3,2,17)

So, “writeln('(2,3,17)');writeln('(3,2,17)');” is the answer.

Proof:

$a^b + b^a$ is prime, which must be an odd number. So either a is 2 or b is 2.

Suppose $a=2$, and $b>3$, b is an odd number. Let consider the following:

By Fermat's Little Theorem, $b^2 \pmod 3$ is always equal to 1.

Since b is odd, We observe that

$$2^1 = 3 \times 0 + 2, 2^3 = 3 \times 2 + 2, 2^5 = 3 \times 10 + 2$$

Suppose $2^{2k-1} = 3n + 2$, multiply 2^2 to both side

We get, $2^{2k+1} = 12n + 8 = 12n + 6 + 2 = 3(4n+2) + 2$, $\pmod 3$ is still 2.

So $2^b \pmod 3 = 2$, add this to b^2 , gives $(2^b + b^2) \pmod 3 = (1+2) \pmod 3 = 0$

So for all odd $b>3$, $2^b + b^2$ is divided by 3, and thus no such pair exists.

Therefore the only solution is $a=2, b=3$ or $a=3, b=2$

Notes: In HKOI, there are questions that require some mathematics skill to solve.

Exercise III:

In Roman numeral representation, almost all numbers are order in descending order,
Except when subtraction is performed.

→ a smaller number followed by a larger number means itself is negative.

```
var rom:array['A'..'Z'] of integer;
    s:string; l,t,i:integer;
begin
rom['I']:=1; rom['V']:=5; rom['X']:=10;
rom['L']:=50; rom['C']:=100; rom['D']:=500;
rom['M']:=1000; readln(s); t:=0;
l:=length(s);
for i:=1 to l do
if i<l then if rom[s[i]]>=rom[s[i+1]] then inc(t,rom[s[i]])
else dec(t,rom[s[i]])
else inc(t,rom[s[i]]);
writeln(t);
end.
```

Exercise IV:

↓,⇒,↑,⇐, this is the sequence of direction starting from the center.

We use a 2d-integer-array to store those numbers and output them finally.

```
Const name='num';
var map:array[0..31,0..31] of integer;
    fi,fo:text;
    lev,a,i,j,g,t,x,y,n:integer;
function f(x,y:integer):integer;
var m,n:integer;
begin
m:=abs(x-a);
n:=abs(y-a);
if m>n then f:=m
else f:=n;
end;

function chk(x,y,lev:integer):boolean;
begin
chk:=(map[x,y]=-1) and (lev=f(x,y));
end;

begin
assign(fi,name+'.in');
assign(fo,name+'.out');
reset(fi);
rewrite(fo);
readln(fi,n);
if n<=9 then g:=2 else g:=3;
fillchar(map,sizeof(map),255);
a:=(n+1) div 2;
x:=a;
y:=a;
t:=0; {Continue}
```

```
{Continue}
lev:=0;
while t<>n*n do
begin
inc(t);
map[x,y]:=t;
if chk(x+1,y,lev) then inc(x)
else if chk(x,y+1,lev) then inc(y)
else if chk(x-1,y,lev) then dec(x)
else if chk(x,y-1,lev) then dec(y)
else begin inc(x);inc(lev);end;
end;

for i:=1 to n do
for j:=1 to n do
if j =n then
writeln(fo,map[i,j])
else write(fo,map[i,j]:g,' ');

close(fi);
close(fo);
end.
```

Notes: fillchar(map,sizeof(map),255) will store all element of array map to be -1.

Exercise V:

By observation of with rows and columns, imagine them as coordinate pairs.

Denote the most inner center point with (x,y) , the value of other point representing the k^{th} shell counting from the outside.

So, to calculate a given point's shell, which is required to find the maximum axial distance between that point and the center one.

For the right hand side, we just calculate the min. instead.

Answer for L.H.S.

```
Var I,j,n,k:integer;
Begin
Readln(n);
K:=(n+1) div 2;
for i:=1 to n do
begin
for j:=1 to n do
write(k- (abs(k-i)+abs(k-j)+abs(abs(k-i)-abs(k-j)))) div 2);
writeln;
end;
end.
```

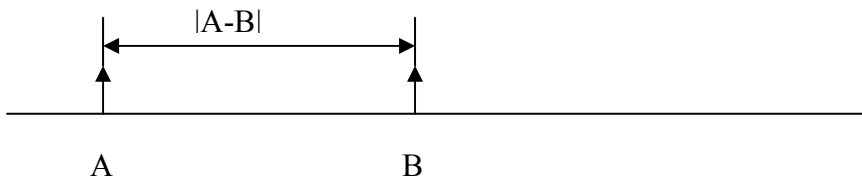
Answer for R.H.S.

```
Var I,j,n,k:integer;
Begin
Readln(n);
K:=(n+1) div 2;
for i:=1 to n do
begin
for j:=1 to n do
write(k- (abs(k-i)+abs(k-j)-abs(abs(k-i)-abs(k-j)))) div 2);
writeln;
end;
end.
```

*Notes: $(A+B+|A-B|) / 2$ always give larger one between A and B.
 $(A+B-|A-B|) / 2$ give the smaller one.*

Number line explanation:

$|A - B|$ is the distance between two points.



Equating max or min of a given number of integers.

Given a function $\text{max2}(a,b)$ which returns the larger one

The function $\text{min2}(a,b)$ is equal to $a+b-\text{max}(a,b)$;

Given a function $\text{max3}(a,b,c)$ which returns the max. one.

$\text{Max3}(a,b,c) = \text{max2}(\text{max2}(a,b),c)$

If $a,b,c > 0$

the min. one of a,b,c can be represented as $a+b+c-\text{max}(a,b,0)-\text{max}(a,c,0)-\text{max}(b,c,0)+\text{max}(a,b,c)$

Otherwise, If $|a|,|b|,|c| < 2^{31} - 1$, $\text{min3}(a,b,c) = -\text{max3}(-a,-b,-c)$;

Otherwise, if any of them is $2^{31} - 1$ or -2^{31} , use this longer equation.

$\text{Min3}(a,b,c) = a+b+c-\text{max}(a,b,-2147483648)-\text{max}(a,c,-2147483648)-\text{max}(b,c,-2147483648)$
 $+ \text{max}(a,b,-2147483648)$

$\text{Max4}(a,b,c,d) = \text{max2}(\text{max2}(\text{max2}(a,b),c),d)$

The function $\text{min4}(a,b,c,d)$ can be represented by $-\text{max}(-a,-b,-c,-d)$. Or

$A+b+c+d+\text{max4}(a,b,c,0)+ \text{max4}(a,b,d,0)+ \text{max4}(a,c,d,0)+\text{max4}(b,c,d,0)-$

$(\text{max4}(a,b,0,0)+\text{max4}(a,c,0,0)+\text{max4}(a,d,0,0)+\text{max4}(b,c,0,0)+\text{max4}(b,d,0,0)+\text{max4}(c,d,0,0))-\text{max4}(a,b,c,d)$;

What should be max5 and min5 ?

Hints of thinking: think of all subsets.

p.s. there are 31 terms for min5 .

Answer for Lesson6_2.doc

Practical Exercise:

1

2

```
var fi:text;
    n,i,s:longint;
begin
assign(fi,'pex1.txt');reset(fi);
readln(fi,n);s:=0;
for i:=1 to n do
begin
readln(fi,n);
inc(s,n);
end;
writeln(s);
close(fi);
end.
{This shows you the feature of for loop}
```

```
var fi,fo:text;
    n,k,i,s:longint;
begin
assign(fi,'pex2.txt');reset(fi);
assign(fo,'pex2o.txt');rewrite(fo);
readln(fi,n);s:=0;
for i:=1 to n do
begin
readln(fi,k);
inc(s,k);
end;
writeln(fo,s/n);
close(fi);close(fo);
end.
```

3

4

```
var fi,fo:text;
    t,h,y,n,i,s:longint;
begin
assign(fi,'pex3.txt');reset(fi);
assign(fo,'pex3o.txt');rewrite(fo);
readln(fi,n);
readln(fi,h);
for i:=2 to n do
begin
readln(fi,y);
if y>h then
begin
y:=y xor h;
h:=y xor h;
y:=y xor h;
end;
while y<>0 do
begin
t:=h;
h:=y;
y:=t mod y;
end;
end;
writeln(fo,h);
close(fi);close(fo);
end.
```

```
var fi,fo:text;
    abcd,t,h,y,n,i,s:longint;
begin
assign(fi,'pex4.txt');reset(fi);
assign(fo,'pex4o.txt');rewrite(fo);
readln(fi,n);
readln(fi,h); abcd:=h;
for i:=2 to n do
begin
readln(fi,y); abcd:=abcd*y;
if y>h then
begin
y:=y xor h;
h:=y xor h;
y:=y xor h;
end;
while y<>0 do
begin
t:=h;
h:=y;
y:=t mod y;
end;
abcd:=abcd div h; h:=abcd;
end;
writeln(fo,abcd);
close(fi);close(fo);
end.
```

Notes: the "begin y:= y xor h ; ... end;" is to swap the value between y and h, it is the best swapping way.

HCM of a,b,c is equal to HCM of (HCM of a and b) and c.

LCM(a,b,c) is equal to LCM (LCM(a,b),c)

We use Euclidean-algorithm to find the HCF first, and then calculate LCM by
 $LCM(a,b) = a*b / HCF(a,b)$

5

```
var fi,fo:text;
    n,s:longint;
begin
assign(fi,'pex5.txt');reset(fi);
assign(fo,'pex5o.txt');rewrite(fo);
s:=0;
while not eof(fi) do
begin
read(fi,n);
inc(s,n);
end;
close(fi);close(fo);
end.
```

6

```
var fi,fo:text;
    s:string;
begin
assign(fi,'pex5.txt');reset(fi);
assign(fo,'pex5o.txt');rewrite(fo);
s:="";
while not eof(fi) do
begin
readln(fi,s);
if s[1]<>' ' then writeln(fo,s);
end;
close(fi);close(fo);
end.
```

7

```
var f1,f2,fo:text;
    t1,t2:longint; z1,z2:boolean;
begin
assign(f1,'pex7_1.txt');reset(f1);readln(f1);
assign(f2,'pex7_2.txt');reset(f2);readln(f2);
assign(fo,'pex7o.txt');rewrite(fo);
readln(f1,t1);readln(f2,t2); z1:=true;
z2:=true;
while not (eof(f1) and eof(f2)) do
begin
if (t1>t2) and z2 then
begin
writeln(fo,t2);
if not eof(f2) then readln(f2,t2)
else begin z2:=false;t2:=2147483647;end;
end
else if (t1<=t2) and z1 then
begin
writeln(fo,t1);
if not eof(f1) then readln(f1,t1)
else begin z1:=false;t1:=2147483647;end;
end
end;
if z1 then writeln(fo,t1);
if z2 then writeln(fo,t2);
close(f1);close(f2);close(fo);
end.
```

8

```
var fi,fo:text;
    n,i,t2:longint; z:boolean;
begin
assign(fi,'pex8.txt');reset(fi);
assign(fo,'pex8o.txt');rewrite(fo);
while not eof(fi) do
begin
read(fi,n);
z:=abs(n)>1;
for i:= 2 to trunc(sqrt(n)) do
z:=z and (n mod i <>0);
if z then writeln(fo,n,' is prime')
else writeln(fo,n,' isn't prime');
end;
close(fi);close(fo);
end.
```

Answer to Lesson7.doc:

Exercise III: 2003 Senior Enumeration

```
var fi,fo:text;
    n,i,j:longint;
begin
assign(fi,'INPUT0.TXT');reset(fi);
assign(fo,'OUTPUT0.TXT');rewrite(fo);
readln(fi,n);
for i:=1 to n do
begin
for j:= 1+sqr(i-1) to sqr(i) - 1 do write(fo,j,' ');
writeln(fo,sqr(i));
end;
close(fi);close(fo);
end.
```

This question requires your observation skill of the first and last number of i^{th} line. $i+(i-1)^2$ and i^2

Exercise IV: 2003 Junior #3 Goldbach's Conjecture

```
var fi,fo:text;
    n,p1,p2,i,j:longint;
    sol,z1,z2:boolean;
begin
assign(fi,'INPUT3.TXT');reset(fi);
assign(fo,'OUTPUT3.TXT');rewrite(fo);
readln(fi,n);
sol:=false;
if odd(n) then
begin
sol:=true;
p1:=2;
p2:=n-p1;
for i:= 2 to trunc(sqrt(p2)) do
sol:=sol and (p2 mod i <>0);
end
else
begin
p1:=n div 2;
p2:=p1;    {Continue on R.H.S.}
```

```
while (p1>=1) and not sol do {Continue from L.H.S.}
begin
z1:=true;
z2:=true;
for i:=2 to trunc(sqrt(p1)) do z1:=z1 and (p1 mod i
<>0);
if z1 then for i:=2 to trunc(sqrt(p2)) do z2:=z2 and (p2
mod i <>0);
if z1 and z2 then sol:=true
else
begin
dec(p1);
p2:=n-p1;
end;
end;
end;
if sol then writeln(fo,p1,' ',p2) else
writeln(fo,'Impossible');
close(fi);close(fo);
end.
```

You must split cases for odd numbers and even numbers.

本題主要考驗同學對質數的認識，能否利用電腦程式分辨出質數與合成數。比賽成績顯示約半數同學懂得使用正確的算法測試數字。

題目給出 N 的範圍是 $[4, 10000000]$ 。由於輸入的數字頗大，同學使用的算法也需要有相當的效率才能在指定時間內完成運行。測試一個數字 N 是否質數的方法其實十分簡單，就是要找出除了 1 和 N 以外，有沒有其他數字能夠整除 N 。很多同學的程式都以 2 至 $N-1$ 作試驗，這個測試範圍可說很大，其實是只要測試 2 至 $\text{trunc}(\text{sqrt}(N))$ 便可(其實更佳的方法是只測試範圍當中的質數)，這個問題我將會在集訓隊的訓練期間與大家討論。

本題並沒有人得滿分，原因是本題的最後一道 testdata 並不單考驗同學編試找質數的能力，同是也測試同學優化算法的能力。雖然哥德巴赫的猜想只適用於大約 2 的偶數，但本題並寫明輸入的數字只有偶數，這便代表 testdata 中有可能有奇數的輸入。有少數同學在比賽中曾向我提問會否有奇數輸入，以及如何處理這種輸入，但我並沒有正面回答。其實，因為質數中有一個是偶數，所以兩個質數的和有可能是奇數的。而測試一個奇數 N 是否兩個質數的和比較簡單，只須知道 $N-2$ 是否質數便可。在 55 個程式中，只有一個程式使用這個方法處理奇數輸入，可惜該程式因其他問題而未能得分。

最多同學犯的錯誤仍然是 overflow。題目已寫明輸入的數字範圍，但有近五份一同學仍然使用 Integer 儲存數字，當輸入的數字大於 32767 時，程式並不能給出正確答案。

另外，部份程式不能正確處理 4 和 5 這兩個輸入。5 是奇數，在上面經已提及，在此不再重覆。4 是一個很特別的數字，因為它的唯一解是 $2+2$ ，是兩個偶數之和，與其他數字不同。某些同學忽略了這個問題，在此損失分數。

少數同學以 hardcode "Impossible" 方法作答本題，遺憾的是本題的 testdata 完全沒有 Impossible 輸出，因為這些程式全部得零分。

最後，值得一提的是題目中寫明輸出 Impossible 時不包括引號，但仍有四位同學沒有依照指示(其中一位得到 90 分)，如果因此而失分便十分不值。

[HKOI 2003 Final Comments: J3 – Goldbach's Conjecture]
[Commented by Unu]

Answer to Lesson7.doc

Practical Exercise

```
{M 1024,0}
var a,b:array[1..10000] of integer;
    cnt:array[1..10000] of byte;
    sum,n,i,j,k:integer;
    done:boolean;
begin
readln(n); sum:=0; done:=false;
fillchar(a,sizeof(a),0);fillchar(b,sizeof(b),0);
fillchar(cnt,sizeof(cnt),0);
i:=1;
while (i<=n) and not done do
begin
read(a[i]);
inc(sum,a[i]);
b[i]:=sum mod n;
inc(cnt[b[i]]);
if (b[i]=0) then
begin
for j:=1 to i-1 do write(a[j],'+');
writeln(a[i], '=',sum,' is divisible by ',n);
done:=true;
end else if cnt[b[i]]>=2 then
begin
k:=1; while (k<i) and (b[k]<>b[i]) do inc(k);
sum:=0;
for j:=k to i do
begin
write(a[j],'+');
inc(sum,a[j]);
end;
writeln(a[i], '=',sum,' is divisible by ',n);
done:=true;
end;
inc(i);
end;
end.
```

This question is modified from a famous IMO question. The original one stated “Please prove that for any N numbers, there exists a subset such that total sum of the integers inside the subset is divisible by n.”

Proof : let those n numbers be $a_1, a_2, a_3, \dots, a_n$

$$\begin{aligned} \text{Let } b_1 &= a_1 \\ b_2 &= a_1 + a_2 \\ b_3 &= a_1 + a_2 + a_3 \\ &\dots \\ &\dots \\ b_n &= a_1 + a_2 + a_3 + \dots + a_n \end{aligned}$$

If any one of b_i is divisible by n, this completes the proof.

Otherwise, suppose all of them is not divisible by n,

We consider the result of $b_i \bmod n$,

Since any number mod n must be $0, 1, 2, \dots, n-1$

Also, since none of them is divisible by n

So the result of $b_i \bmod n$ must be $1, 2, 3, \dots, n-1$

As there are total n numbers, $b_1, b_2, b_3, \dots, b_n$

By Pigeon hole principle, there must be a pair of them leave the same remainder when divided by n.

Let them be b_j and b_k , $j > k$, as they leave the same remainder, so

$b_j - b_k$ must be divisible by n. [Q.E.D.]

Inspired from this proof, we can find the answer by calculating $b_j - b_k$.

Finally, the problem becomes looking for such j and k.

In the code, the integer array **a** stores those n numbers, while array **b** stores the remainder of $b_i \bmod n$, and the byte array **c** stores number of occurrences of each remainder. Sum is to store b_i .

This program shows you a way that processes data while reading data. Since the question only requires one solution, so when one is found, stop the looping.

Notice the comment-liked statement “{M 1024,0}”, it gains more memory at the expense of stack and heap size. In TPW, if it is missed when you compile it, it results in “Compile Error: Data segment too large”.

Also notice the use of the byte array, if cnt is changed to an integer-array, when you try to run the program, it results in “Can’t run program abcd.exe”. As 30000 integers, around 58KB, are very large compared with TPW, TPW can’t handle it and thus an error occurs.

Usual question only test your problem-solving skill, this question test your skills to handle very large set of data, which nearly exceeds the Memory Limit, and test your skills to process data in an efficient way.

In HKOI Final, in order to differentiate a better solution and a worse solution, a memory limit and a run time limit are usually set.

If you code in a bad-style, you will lose marks for most case, because of Memory Limit Exceed (MLE) or Time Limit Exceed (TLE).

All MLE and TLE will be treated as runtime error, and therefore cause zero-mark.