

# TUYỂN TẬP

Đề thi chọn học sinh  
giỏi cấp tỉnh

Tin  
2022 học 2021  
Bậc  
THCS



Trần Hữu Nam

**TUYỂN TẬP**  
**ĐỀ THI CHỌN HỌC SINH GIỎI CẤP TỈNH**  
**MÔN TIN HỌC – BẬC THCS**  
**NĂM HỌC 2021-2022**

1. [CẦN THƠ](#) ..... 3
2. [AN GIANG](#)
3. [BÌNH DƯƠNG](#) x
4. [BÌNH ĐỊNH](#)
5. [ĐÀ NẴNG](#)
6. [ĐẮK NÔNG](#)
7. [ĐỒNG THÁP](#)
8. [HÒA BÌNH](#)
9. [KONTUM](#) x
10. [LONG AN](#)
11. [NGHỆ AN](#)
12. [PHÚ THO](#)
13. [PHÚ YÊN](#)
14. [QUẢNG BÌNH](#)
15. [QUẢNG NINH](#)
16. [QUẢNG TRỊ](#)

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

17.[VĨNH PHÚC](#)

18.[HCM](#) x

19.[HÀ NỘI](#) x

20. TRÀ VINH x

## 1. ĐỀ HSG TIN LỚP 9 TP CẦN THƠ

### Bài 1. Đếm số (6,0 điểm)

Cho một dãy số gồm  $n$  số nguyên.

**Yêu cầu:** Hãy lập trình xác định số lượng các số chia hết cho 3 và không chia hết cho 9 trong dãy số đã cho.

**Dữ liệu:** Vào từ file văn bản **DEMSO.INP** có cấu trúc như sau:

- Dòng đầu ghi một số nguyên dương  $n$  ( $n \leq 10^6$ ).
- Dòng thứ hai ghi  $n$  số nguyên cho biết dãy số, các số trong dãy có giá trị tuyệt đối không vượt quá  $10^9$ .

**Kết quả:** Ghi ra file văn bản **DEMSO.OUT** chứa một số nguyên là kết quả tìm được.

**Ví dụ:**

DEMSO.INP	DEMSO.OUT
6	4
12 3 15 81 6 18	

### Bài 2. Số lẻ (6,0 điểm)

Nam có thói quen thường xuyên thực hiện các phép tính nhanh để rèn luyện khả năng tư duy. Những lần Nam di chuyển bằng xe buýt, nhìn qua cửa sổ xe buýt, Nam thoáng thấy một

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

biển quảng cáo. Nam không chú ý tới nội dung mà chỉ nhớ trên đó có hai số nguyên dương  $a$  và  $b$  ( $a \leq b$ ). Nam có trí nhớ tốt và có khả năng tính nhanh cũng rất tốt, Nam đã tính được số lượng số nguyên trong đoạn  $[a; b]$  có tổng các chữ số là số lẻ. Khi về tới nhà, Nam kiểm tra trên máy tính đã xác định đúng số lượng số nguyên trong đoạn  $[a; b]$  có tổng các chữ số là số lẻ.

**Yêu cầu:** Hãy lập trình xác định kết quả mà Nam đã tính được.

**Dữ liệu:** Vào từ file văn bản **SOLE.INP** gồm một dòng chứa hai số nguyên  $a$  và  $b$  ( $1 \leq a \leq b \leq 10^9$ ).

**Kết quả:** Ghi ra file văn bản **SOLE.OUT** chứa một số nguyên là kết quả Nam đã tính được.

**Ví dụ:**

SOLE.INP	SOLE.OUT
10 20	5

**Giải thích:** Trong ví dụ, các số có tổng các chữ số là số lẻ gồm : 10, 13, 14, 16, 18.

**Ràng buộc:**

- Có ít nhất 70% số test ứng với 70% số điểm của bài thỏa mãn :  $1 \leq a \leq b \leq 10^6$ .
- Có 30% số test còn lại ứng với 30% số điểm của bài thỏa mãn:  $1 \leq a \leq b \leq 10^9$ .

### Bài 3. Kiểu mới (4,0 điểm)

Trong công ty ABC, văn bản gốc chứa các từ chỉ gồm chữ cái La tinh thường và dấu cách. Khi chuyển văn bản này sang chuẩn CamelCase: chữ cái đầu tiên của văn bản có thể bắt đầu bằng chữ cái thường hoặc hoa; chữ cái đầu tiên của mỗi từ tiếp theo được viết hoa và không có dấu cách. Khi Giám đốc điều hành cũ về hưu, Giám đốc điều hành mới quyết định sẽ lưu trữ văn bản theo chuẩn snake\_case. Chuẩn snake\_case được xây dựng theo quy tắc sau: giữa hai từ được cách nhau bởi dấu '\_', tất cả các chữ cái trong văn bản đều là chữ cái thường. Ví dụ, ở chuẩn CamelCase thông tin có dạng abCde thì sang chuẩn snake\_case sẽ có dạng ab\_cde.

**Yêu cầu:** Hãy lập trình đổi thông tin từ chuẩn CamelCase sang chuẩn snake\_case.

**Dữ liệu:** Vào từ file văn bản **KIEUMOL.INP** gồm:

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 100$ )
- Trong  $n$  dòng tiếp theo, mỗi dòng chứa một xâu độ dài không quá 1000 ký tự theo chuẩn CamelCase.

**Kết quả:** Ghi ra file văn bản **KIEUMOL.OUT** các xâu đã cho dưới dạng chuẩn snake\_case, mỗi xâu trên một dòng.

**Ví dụ:**

KIEUMOI.INP	KIEUMOI.OUT
4	camel_case
CamelCase	camel_case
camelCase	to_be_or_not_to_be
toBeOrNotToBe	a_b_c_d_e
ABCDE	

## Bài 4. Quân xe (4,0 điểm)

Trên bàn có một bàn cờ là lưới ô vuông kích thước  $n \times n$ . Các hàng được đánh

	1	2	3	4	5
1				X	
2			X		
3				X	
4					X
5	X				



	1	2	3	4	5
1	X				
2				X	
3			X		
4					X
5		X			

số từ 1 đến  $n$  từ trái qua phải bắt đầu từ 1. Người ta đặt  $n$  quân xe lên bàn cờ sao cho mỗi hàng có đúng một quân xe và mỗi cột cũng có đúng một quân xe. Mỗi quân xe được xác định trên bàn cờ có vị trí ở hàng  $x$ , cột  $y$  (với  $1 \leq x, y \leq n$ ). Khi mang bàn cờ này sang bàn khác, bàn cờ bị xoay đi một góc  $90^\circ$  theo chiều kim đồng hồ.

**Yêu cầu:** Hãy xác định vị trí (hàng, cột) của từng quân xe theo trạng thái mới của bàn cờ và đưa ra các vị trí tìm được theo chiều tăng dần của hàng.

**Dữ liệu:** Vào từ file văn bản **QUANXE.INP** gồm:

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 10^5$ ).
- $n$  dòng tiếp theo, mỗi dòng chứa hai số nguyên dương  $x$  và  $y$  là vị trí hàng  $x$ , cột  $y$  của mỗi quân xe trong bàn cờ lúc đầu.

**Kết quả:** ghi ra file văn bản **QUANXE.OUT** chứa  $n$  dòng, mỗi dòng ghi hai số nguyên là vị trí của quân xe theo trạng thái mới của bàn cờ.

**Ràng buộc:**

- Có ít nhất 70% số test ứng với 70% số điểm của bài thỏa mãn :  $1 \leq n \leq 10^3$ .
- Có 30% số test còn lại ứng với 30% số điểm của bài thỏa mãn :  $1 \leq n \leq 10^5$ .

**Ví dụ:**

QUANXE.INP	QUANXE.OUT
5	1 1
1 4	2 4
2 2	3 3
3 3	4 5
4 5	5 2
5 1	

## HƯỚNG DẪN THUẬT TOÁN

### Bài 1:



Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

Đếm các số chia hết cho 3 mà không chia hết cho 9.

(Lưu ý: phép toán **mod**, **%** có cho **kết quả âm** đối với số âm)

### \* Thuật toán:

1. Dem=0;
2. Nhập n;
3. Với mỗi i chạy từ 1 đến n:
  - 3.1. Nhập x;
  - 3.2. Nếu  $x \bmod 3 = 0$  và  $x \bmod 9 \neq 0$  thì tăng dem;
4. Viết dem;

### \* Code C++:

```
1.  #include <bits/stdc++.h>
2.  using namespace std;
3.  long long n,x,dem = 0;
4.  int main()
5.  {
6.      ios_base::sync_with_stdio(false); cin.tie(NULL);
7.      freopen("DEMSO.INP","r", stdin);
8.      freopen("DEMSO.OUT","w", stdout);
9.      cin>>n;
10.     for(int i = 1; i <= n; i++){
11.         cin>>x;
12.         if(x%3==0 && x % 9 != 0) dem++;
13.     }
14.     cout<<dem;
15.     return 0;
16. }
```

### *Code Python:*

```
1.  import sys
2.  sys.stdin=open("DEMSO.INP")
3.  sys.stdout=open("DEMSO.OUT", "w")
4.  n=int(input())
5.  a=[int(x) for x in input().split()]
6.  dem=0
7.  for x in a:
8.      if x%3==0 and x%9!=0:
9.          dem+=1
10. print(dem)
```

### *Code Pascal:*

```
1.  var n,x,dem,i:longint;
2.  begin
3.      assign(input,'demso.inp');reset(input);
4.      assign(output,'demso.out');
5.      rewrite(output);
6.      readln(n);
7.      dem:=0;
8.      for i:=1 to n do begin
9.          read(x);
10.         if (x mod 3=0)and(x mod 9<>0) then
11.             inc(dem);
12.     end;
13.     writeln(d);
14. end.
```

## Bài 2:

Tóm tắt: Có bao nhiêu số từ a đến b mà tổng các chữ số của nó là số lẻ.

**Ý tưởng 1:** duyệt các số từ a đến b: tính tổng các chữ số của nó  
=> kiểm tra nếu là số lẻ thì tăng biến đếm.

### *\* Thuật toán 1:*

Viết hàm tính tổng các chữ số của số N: **tongcs(N)**

1. **s = 0;**
2. Trong khi  $N > 0$ :
  - 2.1. **s = s + N mod 10;**
  - 2.2. **N = N div 10;**
3. Trả về s;

**Hàm main()** //chương trình chính

1. Nhập a, b;
2. **dem = 0;**
3. Với mỗi x chạy từ a đến b:
  - 3.1. Nếu **tongcs(x) mod 2 = 1** thì tăng dem;
4. Viết dem;

**Cách 2:** Đếm nhanh hơn dựa vào quy luật:

Nhận xét: trong nhóm 10 số 230, 231, 232, 233, 234, 235, 236, 237, 238, 239 có 5 số thỏa mãn có tổng các chữ số là lẻ.

Tổng quát: xét nhóm 10 số liên tiếp:  $\overline{x0}, \overline{x1}, \dots, \overline{x9}$  (x là một số bất kì, chẳng hạn  $x=23$  như ví dụ trên): các số này có tổng các chữ số là **chẵn lẻ xen kẽ nhau**. Bởi vì, nếu  $\text{tongcs}(\overline{x0})$  là lẻ thì  $\text{tongcs}(\overline{x1})$  là chẵn và ngược lại nếu  $\text{tongcs}(\overline{x0})$  là chẵn thì  $\text{tongcs}(\overline{x1})$  là lẻ. Vậy trong nhóm 10 số đó có 5 số có tổng các chữ số là lẻ (tức là một nửa số lượng). Suy ra, từ  $\overline{x0}$  đến  $\overline{y9}$  sẽ có một nửa số lượng số thỏa mãn yêu cầu đề bài, tức là  $\frac{\overline{y9}-\overline{x0}+1}{2}$ .

Gọi x là chữ số đơn vị của a, m là phần các chữ số còn lại, tức là  $a=10m+x$ . Tương tự, ta đặt  $b=10n+y$ .

Chia đoạn  $a \dots b$  thành 3 đoạn:

1.  $10m+x \dots 10m+9$
2.  $10m+10 \dots 10n-1$
3.  $10n \dots 10n+y$

Đoạn 1 và 3 đếm như cách 1 (mỗi đoạn có không quá 10 số) còn đoạn 2 thì lấy một nửa số lượng số trong đoạn.

**Thuật toán 2:**

1. Nhập a, b;
2. dem=0;
3.  $x = a \bmod 10$ ;  $a = a - x$ ;
4.  $y = b \bmod 10$ ;  $b = b - y$ ;
5. Với mỗi t chạy từ a+x đến a+9:
  - 5.1. Nếu  $\text{tongcs}(t) \bmod 2 = 1$  thì tăng dem;
6. Với mỗi t chạy từ b đến b+y:
  - 6.1. Nếu  $\text{tongcs}(t) \bmod 2 = 1$  thì tăng dem;
7.  $\text{dem} = \text{dem} + (b - a - 10) / 2$ ;
8. Viết dem;

### Code C++:

```
1.  #include<bits/stdc++.h>
2.  using namespace std;
3.  int tongcs(int n)
4.  {
5.      int s=0;
6.      while (n>0)
7.      {
8.          s+=n%10;
9.          n/=10;
10.     }
11.     return s;
12. }
13. int main()
14. {
15.     freopen("SOLE.INP", "r", stdin);
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
16.         freopen("SOLE.OUT", "w", stdout);
17.         ios_base::sync_with_stdio(0);
           cin.tie(0);cout.tie(0);
18.         int a,b,dem=0;
19.         cin>>a>>b;
20.         int d,c;
21.         d=(a/10+1)*10;
22.         c=(b/10)*10;
23.         for (int i=a;i<=d-1;i++)
24.             if (tongcs(i)%2!=0) dem++;
25.         dem+=(c-d)/2;
26.         for (int i=c;i<=b;i++)
27.             if (tongcs(i)%2!=0) dem++;
28.         cout<<dem;
29.     }
```

### *Code Python:*

```
import sys
sys.stdin=open("SOLE.INP")
sys.stdout=open("SOLE.OUT", "w")
def tongcs(n):
    s=0
    while n>0:
        s+=n%10
        n//=10
    return s

a,b=[int(x) for x in input().split()]
x=a%10
a-=x
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
y=b%10
b-=y
dem=0
for t in range(a+x, a+10):
    if tongcs(t) % 2 == 1:
        dem+=1
for t in range(b, b+y+1):
    if tongcs(t) % 2 == 1:
        dem+=1
dem+=(b-a-10)//2
print(dem)
```

### *Code Pascal:*

```
1.  var a,b:longint;
2.  function tongcs(x:longint): boolean;
3.  var s:longint;
4.  begin
5.      s:=0;
6.      while x<>0 do begin
7.          s:=s+x mod 10;
8.          x:=x div 10;
9.      end;
10.   exit(s);
11. end;
12. procedure xuli;
13. var x,y,i,dem:longint;
14. begin
15.     dem:=0;
16.     x:=a mod 10;
17.     y:=b mod 10;
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
18.      a:=a-x;
19.      b:=b-y;
20.      if a=b then begin
21.          for i:=a+x to b+y do
22.              if tongcs(i) mod 2 = 1 then
23.                  inc(dem);
24.          writeln(dem); exit;
25.      end;
26.      for i:=a+x to a+9 do
27.          if tongcs(i) mod 2=1 then inc(dem);
28.          for i:=b to b+y do
29.              if tongcs(i) mod 2=1 then inc(dem);
30.          if a+10<b-1 then
31.              dem:=dem+((b-a-10) div 2);
32.          writeln(dem);
33.      end;
34.      begin
35.          assign(input, 'sole.inp'); reset(input);
36.          assign(output, 'sole.out');
37.          rewrite(output);
38.          readln(a, b);
39.          xuli;
40.      end.
```

### Bài 3:

Chuyển kiểu văn bản từ **CamelCase** sang **snake\_case**:

Các kí tự đều chuyển thành chữ thường. Riêng kí tự chữ hoa thì có bổ sung thêm dấu gạch dưới, trừ các kí tự đầu từ.

Ý tưởng:



Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

Khi duyệt gặp chữ cái hoa thì thêm kí tự \_ (trừ trường hợp kí tự đứng **sau dấu cách**). Còn các kí tự đều chuyển thành chữ thường.

S[i] là chữ in hoa khi mã số của nó từ 65 đến 90.

Chuyển kí tự hoa s[i] sang chữ thường:

- Cách 1: mã số chữ thường hơn chữ hoa 32;
  - Pascal/Python: `chr(ord(s[i])+32)`
  - C++: `s[i]+32`
- Cách 2: dùng hàm có sẵn:
  - Pascal: `lowercase(s[i])`
  - C++: `tolower(s[i])` // thư viện `cctype`.
  - Python: `s[i].lower()`

### **\* Thuật toán:**

1. Nhập n;

2. Với mỗi j chạy từ 1 đến n:

2.1. Nhập xâu s;

2.2. p= chữ\_thường(s[1]);

2.3. Với mỗi i chạy từ 2 đến length(s):

2.3.1. Nếu s[i] là chữ hoa và s[i-1] khác dấu cách thì

p=p+ "\_";

2.3.2. p=p + chữ\_thường(s[i]);

## 2.4. Viết p;

### Code C++:

```
1.  #include <bits/stdc++.h>
2.
3.  using namespace std;
4.  int main() {
5.      freopen("KIEUMOI.INP", "r", stdin);
6.      freopen("KIEUMOI.OUT", "w", stdout);
7.      ios_base::sync_with_stdio(0);
        cin.tie(0);cout.tie(0);
8.      long long n;
9.      cin >> n;
10.     cin.ignore();
11.     string s, p;
12.     while(n--) {
13.         getline(cin, s);
14.         p=tolower(s[0]);
15.         for (int i = 1; i < s.size(); i++)
            {
16.             if((s[i]>=65) && (s[i]<=90) && (s[i-1] != ''))
17.                 p+= "_";
18.             p+= tolower (s[i]);
19.         }
20.         cout << p << endl;
21.     }
22.     return 0;
23. }
```

### Code Python:

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
1.  f1=open('kieumoi.inp')
2.  f2=open('kieumoi.out','w')
3.  n=int(f1.readline())
4.  def kt(x):
5.      a=''
6.      for i in x:
7.          if a=='':
8.              a+=i.lower()
9.          elif i.isupper():
10.             a+='_'
11.             a+=i.lower()
12.          elif i=='\n':
13.              return a
14.          else:
15.              a+=i
16.  for i in range(n):
17.      print(kt(f1.readline()),file=f2)
18.  f1.close()
19.  f2.close()
```

### *Code Pascal:*

```
1.  var n,i,j:longint;
2.  s,t:string;
3.  begin
4.      assign(input,'kieumoi.inp');
5.      reset(input);
6.      assign(output,'kieumoi.out');
7.      rewrite(output);
8.      readln(n);
9.      for i:=1 to n do begin
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
10.         readln(s);
11.         t:=lowercase(s[1]);
12.         for j:=2 to length(s) do
13.             if (s[j] in ['A'..'Z']) and (s[j-1]<>#32) then
14.                 t:=t+'_';
15.                 t:=t+lowercase(s[j]);
16.         writeln(t);
17.     end;
18. end.
```

#### Bài 4:

Nhận xét: Ô (x, y) sau khi quay thì trở thành ô (y, n+1-x).

Gọi mảng H[i]: (H[i], i) là tọa độ của 1 quân xe. Hay H[i] là tọa độ hàng của quân xe ở cột i.

Sau khi quay, tọa độ (H[i], i) trở thành (i, n+1-H[i]).

#### \* Thuật toán:

1. Nhập n
2. Với mỗi i chạy từ 1 đến n:
  - 2.1. Nhập x, y;
  - 2.2.  $H[y]=n+1-x$ ;
3. Với mỗi i chạy từ 1 đến n:
  - 3.1. Viết i và H[i];

#### Code C++:

```
#include <bits/stdc++.h>
using namespace std;
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

```
int n, h[100007];
int main()
{
    freopen("QUANXE.inp", "r", stdin);
    freopen("QUANXE.out", "w", stdout);
    cin>>n;
    int x,y,i;
    for(i=1;i<=n;i++)
    {
        cin>>x>>y;
        h[y]=n+1-x;
    }
    for (i=1; i<=n; i++)
        cout<<i<<" "<<h[i]<<endl;
    return 0;
}
```

### *Code Python:*

```
import sys
sys.stdin=open("QUANXE.INP")
sys.stdout=open("QUANXE.OUT", "w")

n=int(input())
h=[0]*(n+1)
for i in range(n):
    x,y=[int(x) for x in input().split()]
    h[y]=n+1-x
for i in range(1,n+1):
    print(i,h[i])
```

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

### *Code Pascal:*

```
var n,x,y,i:longint;
    h:array[1..10000000] of longint;
begin
    assign(input,'QUANXE.INP');reset(input);
    assign(output,'QUANXE.OUT');rewrite(output);
    readln(n);
    for i:=1 to n do
    begin
        readln(x,y);
        h[y]:=n+1-x;
    end;
    for i:=1 to n do writeln(i,' ',h[i]);
end.
```

2.

**Sách bao gồm: đề bài, thuật toán chi tiết, code mẫu bằng 3 NNLT và 1 tài khoản để nộp bài chấm tự động.**

**Link giới thiệu sách: [hoclen.top/go/de221](http://hoclen.top/go/de221)**

**Link đặt sách: [hoclen.top/go/sachde221](http://hoclen.top/go/sachde221)**

**Trang web nộp bài: [tapde221.contest.codeforces.com](http://tapde221.contest.codeforces.com)**

Tài liệu sách: [hoclen.top/sach.html](http://hoclen.top/sach.html)

Zalo: 09 69 156 825

Messenger: [m.me/huunam0](https://m.me/huunam0)

Nếu bạn có đề thi, hãy làm cùng chúng tôi:

<https://hoclen.top/go/gopde>

.