**BAÌ TẬP VỀ NHÀ 7**

***(Hơi khó, các em cố gắng nhé☺)***

**Bài 1**. Cho một dãy gồm một số hữu hạn số nguyên dương. Có thể có những số bằng nhau trong dãy đã cho. Cần tìm cách phân bố các số này vào các dãy con sao cho số lượng của những dãy con như vậy là lớn nhất có thể được, đồng thời mỗi dãy con phải có số lượng phần tử như nhau và tổng các phần tử của một dãy con bất kỳ phải bằng tổng các phần tử của bất kỳ dãy con nào trong số các dãy con còn lại.

**Dữ liệu:** Trong file văn bản EQUAL.INP gồm:

* Dòng đầu tiên chứa số nguyên N (0<N<100) là số lượng phần tử của dãy số.
* Mỗi một trong số N dòng tiếp theo chứa một số của dãy đã cho. Mỗi số như vậy là một số dương không quá 999.

**Kết quả:** Ra file văn bản EQUAL.OUT gồm:

* Dòng đầu ghi số M là số dãy con tìm được
* Mỗi dòng trong số M dò/ ng tiếp theo ghi các phần utử của một trong các dãy con tìm được.

**Ví dụ:**

|  |  |  |  |
| --- | --- | --- | --- |
| EQUAL.INP | EQUAL.OUT | EQUAL.INP | EQUAL.OUT |
| 4  1  2  3  4 | 2  1 4  2 3 | 4  1  1  1  1 | 4  1  1  1  1 |

**Bài 2.** Cho N số tự nhiên từ 1, 2, .., N( N<=9). Tìm các số có K chữ số khác nhau được chọn từ dãy N số trên. ( 1<=k<=N<=9).

Dữ liệu vào từ file: CHONSO.INP gồm 2 số N và K trên 1 dòng

Dữ ra ghi vào file: CHONSO.OUT các số tìm được, mỗi số trên 1 dòng sắp xếp từ bé đến lớn.

Ví dụ:

|  |  |
| --- | --- |
| CHONSO.INP | CHONSO.OUT |
| 3 2 | 12  13  21  23  31  32 |

**Bài 3.** Với một hệ điều hành hoạt động bình thường ta có thể mở được nhiều của sổ. Mỗi một cửa sổ là một hình chữ nhật chứa nhiều ô vuông nhỏ (1x1). Những cửa sổ mới mở, tùy thuộc vào vị trí và khích thước của nó, có thể bao (một phần hay toàn thể) cửa sổ được mở trước. Để đóng cửa sổ chúng ta có thể nháy chuột vào ô vuông nhỏ phía trên bên phải có thể nhìn thấy được của cửa sổ (là tức không bị của sổ mở sau, là cửa sổ chưa đóng, che khuất).

Viết chương trình tính số nhỏ nhất các lần nháy chuột để đóng của sổ được mở đầu tiên.

***Dữ liệu vào từ file CUASO.INP***

Dòng đầu tiên chứa số nguyên N là số các cửa sổ, 1<N<100. Mỗi dòng trong N dòng tiếp theo chứa 4 số nguyên R1, S1, R2, và S2, 1<R1<R2<10000, 1<S1<S2<10000. Các số R1 và S1; R2 và S2 tương ứng là hàng ngang và cột dọc của màn hình ô vuông trên trái và ô vuông dưới phải của một cửa sổ. Các cửa sổ được mở theo cùng một trật tự theo trật tự xuất hiện.Màn hình gồm các hàng ngang và các cột dọc của các ô vuông nhỏ. Các hàng ngang được đánh số từ trên xuống dưới, các cột dọc từ trái sang phải và ô vuông trên trái trên màn hình ở trong hàng ngang và cột dọc đầu tiên.

***Kết quả ghi ra file CUASO.OUT***

Chỉ một dòng duy nhất là số lần kích chuột

Ví dụ:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **cuaso.inp** | **cuaso.out** | **cuaso.inp** | **cuaso.out** | **cuaso.inp** | **cuaso.out** |
| 3  3 1 6 4  1 2 4 6  2 3 5 5 | 3 | 3  4 1 6 3  2 2 5 5  1 4 3 6 | 3 | 3  3 3 4 4  1 1 2 2  5 5 6 6 | 1 |

**Bài 4.** Để quản lý các khách mời tới dự hội tr­­ờng nhân dịp kỷ niệm 40 năm thành lập tr­­ờng chuyờn Phan Bội Chõu. Ban tổ chức cần đánh mã số cho từng khách mời. Để gợi nhớ mã số cho từng ng­­ời. Ban tổ chức quyết định đánh mã số theo n­­ớc đi con mã của cờ vua trên các phím bấm chữ số của điện thoại di động.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |
|  | 0 |  |

Con mã di chuyển bằng một b­­ớc trên bảng các chữ số này nh­­ sau :

* *Từ ô số 1 có thể di chuyển tới các ô số 6 hoặc 8 và ng­­ợc lại,*
* *Từ ô số 2 có thể di chuyển tới các ô số 7 hoặc 9 và ng­­ợc lại,*
* *Từ ô số 3 có thể di chuyển tới các ô số 4 hoặc 8 và ng­­ợc lại,*
* *Từ ô số 4 có thể di chuyển tới các ô số 3, 9 hoặc 0 và ng­­ợc lại,*
* *Từ ô số 6 có thể di chuyển tới các ô số 1, 7 hoặc 0 và ng­­ợc lại.*

Mã số của một ng­­ời là một số có N chữ số, không bắt đầu bằng 0 và hai chữ số đứng kề nhau t­ơng ứng với hai ô mà thì con mã có thể di chuyển từ ô này sang ô kia bằng một b­­ớc. Ví dụ với N = 4 thì số 1672 sẽ là mã hợp lệ, còn số 1860 không là một mã hợp lệ bởi vì từ ô 8 con mã không di chuyển đ­­ợc tới ô 6 bằng 1 b­­ớc. Để thuận lợi cho việc quản lý Ban tổ chức muốn mã số không quá dài, vì vậy Ban tổ chức cần biết với độ dài của mã N cho tr­­ớc thì có thể mã số tối đa đ­­ợc cho bao nhiêu ng­­ời.

***Yêu cầu***: Với N cho tr­­ớc, hãy tìm số l­­ợng lớn nhất S các mã số có thể có gồm N chữ số (1 < N < 24).

***Dữ liệu*** : Vào từ file văn bản ID.INP trong đó chứa một số duy nhất N.

***Kết quả*** : Đ­a ra file văn bản ID.OUT trong đó ghi ra số S tìm đ­­ợc.

***Ví dụ*** :

|  |  |
| --- | --- |
| ID.INP | ID.OUT |
| 3 | 40 |

**Bài 5.** Cho lưới ô vuông cấp NxM. Trên mỗi ô [i,j] của lưới ghi một số nguyên a[i,j]. Hai ô được gọi là liên thông trực tiếp nếu nó chung cạnh và giá trị tuyệt đối của tổng 2 số ghi trên 2 ô đó  là số chẵn. Hãy lập trình giải quyết các công việc sau:

         Cho biết lưới ô vuông đó có bao nhiêu vùng liên thông( vùng liên thông gồm các ô liên thông trực tiếp hoặc liên thông qua một số trung gian nào đó)

**Dự liệu vào**:  Cho trong tệp văn bản LUOI.txt có cấu trúc như sau:

          – Dòng đầu tiên ghi hai số nguyên dương n, m là kích thước của lưới

          – Dòng thứ i trong N dòng tiếp theo ghi M số nguyên dương là các số ghi trên dòng thứ i của lưới theo thứ tự từ trái qua phải.

**Kết quả**:  Đưa ra tệp văn bản LUOI.out, có cấu trúc như sau:

        ghi số S là số vùng liên thông của lưới.

*Ví dụ*

|  |  |
| --- | --- |
| LUOI.inp | LUOI.out |
| 4 5  0 1 3  1   5  1 1 5  7   8  2 2 4  1   5  0 5 9 10  2 | 6 |

**Bài 6.** Viết ch­­ơng trình tìm tất cả các cách phân tích số tự nhiên N ( 1<N <101) thành tổng các số nguyên tố không v­­ợt qúa N. Hai cách phân tích chỉ khác nhau bởi thứ tự các số hạng đ­­ợc xem là một cách.

**Dữ liệu**: Nhập từ file văn bản NT.INP chỉ một dòng duy nhất ghi một số N.

**Kết quả** : Xuất ra file văn bản NT.OUT số cách có thể phân tích đ­­ợc.

**Ví dụ:** Với N = 9 ta có 4 cách phân tích :

- Cách 1: 9 = 2+2+2+3.

- Cách 2: 9= 2+2+5.

- Cách 3: 9 = 2+7.

- Cách 4: 9= 3+3+3.

Do đó ta trả lời: có 4 cách phân tích (không cần trình bày cách phân tích ).

|  |  |
| --- | --- |
| NT.INP | NT.OUT |
| 9 | 4 |

**Bài 7.** Xét xem 1 số tự nhiên N ( 1<= N <= 100) có phải là số thuộc dãy fibonacci hay không và phân tích nó dưới dạng tổng của các số fibonacci ( Các số có thể lặp lại)

|  |  |  |
| --- | --- | --- |
| PT\_FIB.INP | PT\_FIB.OUT | Giải thích |
| 4 | NO  4 | Có 4 cách phân tích  1+1+1+1  1+1+2  1+3  2+2 |
| 5 | YES  6 | Có 6 cách phân tích  1+1+1+1+1  1+1+1+2  1+1+3  1+2+2  2+3  5 |

**Bài 8.** Số siêu nguyên tố có n chữ số được định nghĩa như sau. Khi ta bỏ 0, 1, …, n – 1 chữ số bên phải thì nó vẫn là số nguyên tố. 7331 là một số siêu nguyên tố có 4 chữ số vì 7331, 733, 73, 7 đều là những số nguyên tố. Viết chương trình in ra tất cả các số siêu nguyên tố có n chữ số với n <= 8. Lưu ý 1 không phải là số nguyên tố

Tên file: sprime.\*

Input: sprime.inp

Gồm 1 số nguyên dương n <= 8.

Output: sprime.out

Gồm nhiều dòng, mỗi dòng biểu diễn 1 số siêu nguyên tố theo thứ tự tăng dần.

Ví dụ:

Sprime.inp

4

Sprime.out

2333

2339

2393

2399

2939

3119

3137

3733

3739

3793

3797

5939

7193

7331

7333

7393

**Bài 9.** Cho một số tự nhiên N ≤ 9. Giữa các số từ 1 đến N hãy thêm vào cácdấu + và - sao cho kết quả thu được bằng 0. Hãy viết chương trình tìmtất cả các khả năng có thể.   
*Dữ liệu vào:* Lấy từ file văn bản ZERO.INP với một dòng ghi số N.   
*Dữ liệu ra:* Ghi vào file văn bản có tên ZERO.OUT: ghi sốlượng kết quả tìm được.   
**Ví dụ**   
