**Bài 1. BỮA TIỆC KHIÊU VŨ Tên file: DANCING.PAS**

Có N là chàng trai và N cô gái tham gia một bữa tiệc khiêu vũ. Chiều cao của họ đã được đo và đưa vào một danh sách. Mỗi chàng trai sẽ chỉ nhảy với một cô gái và ngược lại. Tức là mỗi người chỉ có nhiều nhất một bạn nhảy.

Hai cặp trai gái sẽ không nhảy với nhau nếu như họ có cùng chiều cao. Hãy xác định tối đa các cặp có thể được khiêu vũ với nhau.

INPUT: Đọc từ file **DANCING.INP**:

* Dòng đầu tiên chứa một số nguyên dương N (1 ≤ N ≤ 100.000).
* Dòng thứ hai chứa N số nguyên có giá trị tuyệt đối thuộc [1500, 2500]. Các giá trị tuyệt đối của các số nguyên thể hiện chiều cao của những chàng trai (tính bằng milimet). Chiều cao dương thể hiện chàng trai muốn nhảy với cô gái cao hơn minh, chiều cao âm thể hiện chàng trai muốn nhảy với cô gái thấp hơn mình.
* Dòng thứ ba chứa N số nguyên có giá trị tuyệt đối thuộc [1500, 2500]. Các giá trị tuyệt đối của các số nguyên thể hiện chiều cao của những cô gái (tính băng milimet). Chiều cao dương thể hiện cô gái muốn nhảy với chàng trai cao hơn minh, chiều cao âm thể hiện cô gái muốn nhảy với chàng trai thấp hơn mình.

OUTPUT: Ghi ra file văn bản **DANCING.OUT** chứa một số nguyên dương duy nhất là số lượng lớn nhất các cặp nhảy có thể.

Ví dụ

|  |  |  |
| --- | --- | --- |
| DANCING.INP | DANCING.INP | DANCING.INP |
| 1  -1800  1800 | 1  1700 2000  -1800 -1800 | 2  -1800 -2200  1900 1700 |
| DANCING.OUT | DANCING.OUT | DANCING.OUT |
| 0 | 1 | 2 |

**Bài 2. Đếm tần số file: COUNT.PAS**

Cho trước một dãy N phần tử A1, A2, …, AN và một số nguyên dương k.

Bạn được phép chọn bất kỳ một phần tử nào đó trong dãy và tăng nó lên 1 đơn vị. Bạn được thực hiện phép thay đổi trên không quá k lần (tức là có thể không cần dùng hết k lần).

Nhiệm vụ của bạn là tìm ra phần tử xuất hiện nhiều nhất trong dãy sau khi đã thực hiện các phép thay đổi như trên.

INPUT: COUNT.INP

* Dòng đầu tiên chứa hai số nguyên dương N và k (1 ≤ n ≤ 105; 0 ≤ k ≤ 109)
* Dòng thứ hai chứa N số nguyên, mỗi số có giá trị tuyệt đối nhỏ hơn hoặc bằng 109.

OUTPUT: COUNT.OUT

* Chứa hai số nguyên dương *t* và *c*, trong đó *t* là số lần xuất hiện nhiều nhất của *c* trong dãy *A*,nếu có nhiều phương án thay đổi thì hãy đưa ra phương án có giá trị bé nhất xuất hiện nhiều lần nhất.

Ví dụ:

|  |  |
| --- | --- |
| **COUNT.INP** | **COUNT.OUT** |
| 5 3  6 3 4 0 2 | 3 4 |
| 3 4  5 5 5 | 3 5 |
| 5 3  3 1 2 2 1 | 4 2 |

\* Giải thích ví dụ 2:

Có 2 phương án biến đổi:

- Không thay đổi gì ta được kết quả là 3 5

- Thay đổi cả 3 giá trị được kết quả là 3 6

Tuy nhiên 5 nhỏ hơn 6 nên kết quả là 3 5

**Bài 3: DVONIZ**

Một dãy độ dài được coi là *interesting*, nếu tổng phần tử đầu tiên nhỏ hơn hoặc bằng và tổng phần tử tiếp theo nhỏ hơn hoặc bằng .

Cho dãy độ dài . Với tất cả mọi vị trí trong dãy , bạn cần tìm một dãy *interesting* dài nhất bắt đầu từ vị trí .

**INPUT: DVONIZ.INP**

* Dòng đầu tiên gồm 2 số nguyên và .
* dòng sau mỗi dòng ghi 1 số nguyên thể hiện số thứ trong dãy . Các số là nguyên dương và tổng của chúng không vượt quá .

**OUTPUT:** **DVONIZ.OUT**

* Gồm dòng, dòng thứ thể hiện dãy *interesting* dài nhất bắt đầu ở vị trí thứ trong dãy.

**Ví dụ:**

|  |  |  |
| --- | --- | --- |
| input  5 10000  1  1  1  1  1 | input  5 9  1  1  10  1  9 | input  8 3  1  1  1  1  1  1  1  1 |
| output  4  4  2  2  0 | output  2  0  0  2  0 | output  6  6  6  4  4  2  2  0 |

|  |  |
| --- | --- |
| **Summary** | |
| Program | familiar.\* |
| Input | familiar.inp |
| Output | familiar.out |
| Time/test | 1 giây |

# Bài 4. GƯƠNG MẶT THÂN QUEN

"*Gương mặt thân quen"* là một chương trình giải trí khá nổi tiếng trên VTV3. Trong chương trình này, mỗi thí sinh sẽ bắt chước giọng hát của một ca sỹ nổi tiếng nào đó và trên cơ sở đó bạn giám khảo sẽ cho điểm từng thí sinh.

Có tất cả N thí sinh tham gia thi. Cuộc thi được diễn ra trong nhiều vòng thi khác nhau. Mỗi vòng thi, thí sinh tốt nhất sẽ được N điểm, thí sinh tốt thứ nhì được N-1 điểm, thí sinh tốt thứ ba được N-2 điểm, ..., thí sinh tốt thứ N được 1 điểm. Điểm của mỗi vòng thi của từng thí sinh được cộng lại, sau vòng thi cuối cùng thí sinh nào được nhiều điểm nhất sẽ giành chức vô địch. Tất nhiên, nếu có nhiều thí sinh cùng đạt nhiều điểm nhất thì tất cả họ đều giành chức vô địch.

Chỉ còn một vòng thi nữa là cuộc thi kết thúc. Hiện tại điểm tổng của các thí sinh là . Hỏi rằng có bao nhiêu thí sinh có quyền hy vọng rằng mình sẽ đạt chức vô địch sau vòng thi cuối cùng?

**Input:**

* Dòng đầu tiên chứa số nguyên N () là số lượng thí sinh tham gia thi
* N dòng tiếp theo, mỗi dòng ghi một số nguyên *ai* () là số điểm của các thí sinh trước vòng thi cuối cùng

**Output:** Một số nguyên duy nhất là số lượng thí sinh có thể đạt được chức vô địch sau vòng thi cuối cùng

**Example:**

|  |  |  |
| --- | --- | --- |
| Input | Output |  |
| 5  15  14  15  12  14 | 4 |  |
| 3  8  10  9 | 3 |  |

**Bài 5. Sắp xếp và tìm kiếm nhị phân**

Sắp xếp dãy điểm số theo thứ tự giảm dần.

Xét tay đua thứ K.

+ Nếu cho tay đua này về nhất thì số điểm là:

* A[k] + N > A[k+1] + N – 1 (số điểm cao nhất mà các tay đua phía sau có thể dành được), như vậy đã loại bỏ hoàn toàn các tay đua từ vị trí k + 1 đến vị trí N
* Với số điểm A[k] + N xét với những tay đua phía trên (từ 1 -> K – 1): Tình huống tốt nhất cho người thứ K là những người xếp trên thi đấu tồi nhất, tức là A[1] về bét, A[2] về gần bét, ..., cứ như vậy A[k – 1] về vị trí N – k + 1. Khi đó nếu A[k] + N ≥ A[1] + 1 thì tay đua thứ K sẽ có thể dành chức vô địch. Còn nếu A[k] + N < A[1] + 1 thì chắc chắn sẽ không thể vô địch

+ Nếu người thứ K không thể dành chức vô địch hiển nhiên những người xếp dưới cũng sẽ không thể, ta sẽ xét những người thứ 1 đến k – 1. Và lặp lại quá trình trên.

+ Nếu người thứ K có thể danh chức vô địch, hiển nhiên những người xếp trên cũng sẽ có thể vô địch (khi đó cập nhật kết quả bằng K), ta sẽ xét những người từ thứ k + 1 đến N. Và lặp lại quá trình trên.

Để việc tìm K được đơn giản thì tốt nhất là dùng tìm kiếm nhị phân.

### **Bài 6. Kết bạn (Đề thi Olympic các tỉnh duyên hải - 2010)**

Theo quan niệm của người Á Đông cổ, mỗi cá nhân khi sinh ra đều ứng với một ngôi sao, được gọi là sao chiếu mệnh. Các hoạt động của cá nhân đều bị chi phối bởi ngôi sao này, kể cả quá trình kết bạn – hẹn hò. Theo thuyết Âm dương – Ngũ hành, hai người chỉ có thể tạo lập mối quan hệ bền vững khi các sao chiếu mệnh của họ không có các thuộc tính tương khắc. Qua hàng nghìn năm quan sát và chiêm nghiệm, các chiêm tinh gia đã ghi nhận được *n* sao và hầu hết các tính chất tương sinh – tương khắc giữa chúng. Để có thể nhanh chóng đáp ứng nhu cầu kiểm tra độ tương hợp của các sao, hiệp hội **ABS** (***A****ssociation of* ***B****roker for* ***S****ingle*) tạo lập cơ sở dữ liệu ghi nhận tính chất của tất cả các sao đã khảo sát được. Trong cơ sở dữ liệu này, các sao được đánh số từ 1 tới *n*; sao thứ *i* có một giá trị *si* thể hiện khả năng thích nghi của sao gọi là độ thích nghi. Hai sao khác nhau có thể có cùng độ thích nghi. Thông qua độ thích nghi của các sao, người ta xác định khả năng tương hợp của chúng. Khả năng tương hợp của 2 sao được tính bằng tổng 2 độ thích nghi của chúng.

***Bài toán:*** Cho số nguyên dương *n*, dãy *s*1, *s*2, …, *sn* là độ thích nghi của các sao và số nguyên *B*. Hãy xác định số lượng các cặp sao (*i*, *j*) với *i* < *j* và *si* + *sj* = *B*.

***Dữ liệu:*** Vào từ file văn bản FRIEND.INP:

Dòng đầu tiên ghi 2 số nguyên *n*, *B* (2 ≤ *n* ≤ 105, |*B*| ≤ 109),

Mỗi dòng trong *n* dòng tiếp theo ghi một số nguyên là độ thích nghi của một sao, độ thích nghi có trị tuyệt đối ≤ 109.

***Kết quả:*** Đưa ra file văn bản FRIEND.OUT một số nguyên – số lượng cặp sao có độ tương hợp *B* tìm được.

***Ví dụ:*** Trong 5 sao với độ thích nghi 3, 5, 6, 5, 3 có 4 cặp có khả năng tương hợp bằng 8.

|  |  |  |
| --- | --- | --- |
| **FRIEND.INP** |  | **FRIEND.OUT** |
| 5 8  3  5  6  5  3 |  | 4 |
|  |

**Bài 7. KHAI THÁC DẦU** *Tên chương trình*: OIL.???

Hãng *RectanGas* chuyên khai thác dầu từ thời xa xưa và cho đến nay vẫn là một hãng dẫn đầu không có đối thủ. Các chuyên gia tiếp cận thị trường cho rằng thành công của hãng gắn liền với vị Giám đốc điều hành nổi tiếng. Về ông này chúng ta phải nói kỹ hơn một chút.

Không rõ học vấn của ông có được từ đâu và như thế nào, nhưng khi động chạm đến hình học, người ta chỉ thấy ông nói tới hình chữ nhật mà thôi. Hơn thế nữa, người ta có cảm giác là với ông, đã là hình chữ nhật là dứt khoát cạnh phải song song với trục tọa độ! Ông điều hành việc khai thác dầu như sau: Chọn trên bản đồ 4 lô (lưu ý, mỗi lô là một hình vuông, toàn bộ vùng có dầu được chia thành các lô) là 4 lô góc của một hình chữ nhật có cạnh song song với trục tọa độ, ra lệnh lắp các dàn khoan ở đó và mau chóng hút sạch trữ lượng dầu tại các lô đó.

Hãng *RectanGas* gần hoàn thành kế hoạch quý 2 năm 2010. Chỉ còn phải khai thác nốt ***S*** barrel (thùng) nữa là xong. Cần phải biết rằng Hãng không được khai thác nhiều hơn, nếu không sẽ vi phạm luật Chống độc quyền. Còn khai thác ít hơn ư? Bạn có muốn mất việc không đấy? Giám đốc điều hành là một người rất dứt khoát. Phòng thăm dò đã lên bản đồ trữ lượng của một vùng ***n***×***m*** lô (dĩ nhiên là hình chữ nhật cạnh song song với trục tọa độ, ***n*** dòng, ***m*** cột). Vấn đề là phải chọn 4 lô dặt dàn khoan khai thác.

***Yêu cầu:*** Cho ***n***, ***m*** và ***S*** (2 ≤ ***n***, ***m*** ≤ 1000, 0 ≤ ***S*** ≤ 107), trữ lượng ***aij*** (0 ≤ ***aij*** ≤ 106) trong mỗi lô (***i***,***j***). Hãy chỉ ra tọa độ lô trên trái và dưới phải của hình chữ nhật xác định vị trí đặt dàn khoan khai thác. Nếu không tồn tại cách khai thác thì đưa ra số -1.

***Dữ liệu:*** Vào từ file văn bản OIL.INP:

* Dòng đầu tiên chứa 3 số nguyên ***n***, ***m*** và ***S***,
* Dòng thứ ***i*** trong ***n*** dòng sau chứa m số nguyên ***ai1***, ***ai2***, . . ., ***aim***.

***Kết quả:*** Đưa ra file văn bản OIL.OUT bốn số nguyên xác định tọa độ lô trên trái và tọa độ lô dưới phải hoặc đưa ra một số -1.

***Ví dụ:***

|  |  |  |
| --- | --- | --- |
| OIL.INP |  | OIL.OUT |
| **3 3 16**  **1 3 5**  **2 4 8**  **6 9 7** |  | **1 1 2 3** |
|  |  |

 x08Int\_A\_100508\_H

Solution

* Sinh mảng Sum[d,j] = A[i,j] + A[k,j]
* Xét dòng i của bảng Sum: tìm 2 phần tử có tổng bằng K tại dòng này

**Bài 8. Số ngẫu nhiên. Tên file chương trình là RANNUM.CPP**

Cho dãy số nguyên a1,a2,..,an. Số ai được gọi là số k-ngẫu nhiên của dãy nếu trong k số hạng liên tiếp bất kì của dãy đều có ít nhất một số hạng bằng ai và k là số nguyên nhỏ nhất thỏa mãn điều kiện này.

Ví dụ: Dãy 1,2,3,1,2,2. Số 1 là số 3-ngẫu nhiên; số 2 là số 3-ngẫu nhiên; số 3 là số 4-ngẫu nhiên.

**Yêu cầu:** Tìm k nhỏ nhất để trong dãy có số k – ngẫu nhiên.

**Input:** Cho trong tệp văn bản **RANNUM.INP** như sau:

* Dòng đầu ghi số nguyên dương n (N<=10^5)
* Dòng thứ 2 ghi n số nguyên a1,a2,..an.(|ai|<=10^3).

**Output:** Ghi trong tệp văn bản RANNUM.OUT gồm một số k tìm được thỏa mãn yêu cầu bài toán.

Ví dụ:

|  |  |
| --- | --- |
| **RANNUM.INP** | **RANNUM.OUT** |
| 6  1 2 3 1 2 2 | 3 |

**Bài 9. Dãy con Tên file chương trình là Sub.pas**

Cho một dãy số nguyên dương a1,a2,...,aN (10 < *N* < 105), ai <=109 với mọi i=1..N và một số nguyên dương S (*S* < 109).

***Yêu cầu :*** Tìm độ dài nhỏ nhất của dãy con chứa các phần tử liên tiếp của dãy mà có tổng các phần tử lớn hơn hoặc bằng S.

***Dữ liệu vào:*** Đọc từ file **SUB.INP**gồm 2 dòng, dòng 1 chứa N và S ở dòng đầu. Dòng 2 chứa các phần tử của dãy.

***Dữ liệu ra:*** Kết quả ghi vào file **SUB.OUT**, chứa độ dài của dãy con tìm được.

***Ví dụ :***

|  |  |
| --- | --- |
| **SUB.INP** | **SUB.OUT** |
| 10 17  5 1 3 5 10 7 4 9 2 8 | 2 |

**Bài 10. Đếm tam giác. Tên file chương trình là TRIANGLE.PAS**

Cho 3 dãy số dương A, B, C cùng có N phần tử. Hãy đếm xem có bao nhiêu bộ 3 số A[i], B[j] và C[k] mà 3 số này là 3 cạnh của 1 tam giác.

**Dữ liệu vào:** từ file TRIANGLE.INP với cấu trúc:

* Dòng đầu chứa số nguyên n (n <= 1000)
* Dòng thứ hai chứa các số A1, A2, ..., An.
* Dòng thứ ba chứa các số B1, B2, ..., Bn.
* Dòng thứ tư chứa các số C1, C2, ..., Cn.

Các số ai, bi, ci đều không vượt quá 104 và được ghi cách nhau bởi dấu cách.

**Dữ liệu ra:** file văn bản TRIANGLE.OUT gồm một số S duy nhất là số lượng bộ ba số tìm được.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TRIANGLE.INP** | **TRIANGLE.OUT** |  | **TRIANGLE.INP** | **TRIANGLE.OUT** |
| 2  2 3  3 1  4 7 | 2 |  | 3  2 3 1  4 4 9  8 5 2 | 8 |

**Bài 11. CẮT GỖ Tên file: CATGO1.CPP**

Nông dân Jonh cần phải cắt M mét gỗ để đem về làm chuồng bó cho đàn bò của mình. Bác có một cái máy cắt gỗ mới, vì vậy mà công việc chặt gỗ cũng chở lên dễ dàng hơn. Tuy nhiên, bác Jonh chỉ được phép để cắt các cây gỗ thành một hàng duy nhất.

Máy cắt gỗ của bác Jonh hoạt động như sau: bác thiết lập một tham số chiều cao H (tính bằng mét), và máy sẽ cắt toàn bộ các cây có chiều cao lớn hơn H (tất nhiên, cây không cao hơn H mét vẫn còn nguyên vẹn). Sau đó bác chỉ chọn những phần cây bị cắt đứt. Ví dụ, nếu hàng cây chứa cây với chiều cao là 20, 15, 10, và 17 mét, và bác Jonh thiết lặp chiều cao là 15 mét, chiều cao các cây còn lại sau khi cắt sẽ là 15, 15, 10, và 15 mét, khi đó bác Jonh sẽ nhặt 5 mét của cây đầu tiên và 2 mét của cây thứ tư và có tổng số là 7 mét gỗ.

Bác Jonh là người tiết kiệm, vì vậy bác không muốn cắt gỗ nhiều hơn cần thiết. Vì vậy bác muốn thiết lập cho máy của mình chiều cao H càng cao càng tốt. Trợ giúp bác Jonh tìm số nguyên chiều cao tối đa của chiều cao H đề bác vẫn có thể cắt ít nhất là M mét gỗ.

**INPUT: CATGO1.INP**

* Dòng đầu tiên chứa 2 số nguyên dương N (1 ≤ N ≤ 106) là số lượng cây gỗ và M (1 ≤ M ≤ 2\*109) là tổng độ dài mét gỗ cần khai thác.
* Dòng thứ hai chứa N số nguyên dương (nhỏ hơn 109) là chiều cao của mỗi cây gỗ (tính bằng mét). Dữ liệu vào luôn đảm bảo là bác Jonh có thể cắt được M mét gỗ.

**OUTPUT: CATGO1.OUT**

* Một dòng duy nhất là chiều cao H lớn nhất cần thiết lập.

Ví dụ:

|  |  |
| --- | --- |
| **CATGO1.INP** | **CATGO1.INP** |
| 4 7  20 15 10 17 | 5 20  4 42 40 26 46 |
| **CATGO1.OUT** | **CATGO1.OUT** |
| 15 | 36 |

**BÀI 12. ĐÓNG GÓI SẢN PHẨM Tên file: ZXY.CPP**

Ở đầu ra của một dây chuyền sản xuất trong nhà máy ZXY có một máy xếp tự động. Sau khi kết thúc việc gia công trên dây chuyền, các sản phẩm sẽ được xếp vào các hộp có cùng dung lượng M. Sản phẩm rời khỏi dây chuyền được xếp vào hộp đang mở (khi bắt đầu ca làm việc có một hộp rỗng được mở sẵn) nếu như dung lượng của hộp còn đủ để chứa sản phẩm. Trong trường hợp ngược lại, máy sẽ tự động đóng nắp hộp hiện tại, cho xuất xưởng rồi mở một hộp rỗng mới để xếp sản phẩm vào. Trong một ca làm việc có n sản phẩm đánh số từ 1 đến n theo đúng thứ tự mà chúng rời khỏi dây chuyền. Sản phẩm thứ i có trọng lượng là ai, i = 1, 2, …, n. Ban Giám đốc nhà máy qui định rằng sản phẩm xuất xưởng của mỗi ca làm việc phải được xếp vào trong không quá k hộp.

Yêu cầu: Hãy giúp người quản đốc của ca làm việc xác định giá trị M nhỏ nhất sao cho số hộp mà máy tự động cần sử dụng để xếp dãy n sản phẩm xuất xưởng của ca không vượt quá số k cho trước.

Dữ liệu: Vào từ file văn bản ZXY.INP:

• Dòng đầu tiên chứa hai số nguyên n và k, (1­ <= k <=­ n <=­ 15000);

• Dòng thứ i trong n dòng tiếp theo chứa số nguyên dương ai (ai <= ­30000), i =1, 2, …, n.

Các số trên một dòng cách nhau ít nhất một dấu cách.

Kết quả: Ghi ra file ZXY.OUT một số nguyên duy nhất là dung lượng của hộp.

Ví dụ:

