|  |  |
| --- | --- |
| HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  **KHOA CÔNG NGHỆ THÔNG TIN I** | **CONTEST LẬP TRÌNH TRỰC TUYẾN**  Học kỳ: **3 –** Năm học: **2023 – 2024** |
| Lớp: **Thuật toán và ứng dụng cơ bản – 2024**  Thời gian thi: **120 phút** | Chủ đề: **Luyện tập Lập trình cơ bản** |

**ĐỀ BÀI**

**BÀI 1. Ctest001. TÌM SỐ LỚN NHẤT**

Cho một dãy vô hạn các số nguyên không quá 18 chữ số, các số có thể cách nhau một vài khoảng trống, có thể xuống dòng. Hãy tìm số nguyên lớn nhất và in ra màn hình. Chú ý là nếu số đó xuất hiện nhiều lần thì vẫn chỉ in một lần

***Sinh viên cần tư đọc hết luồng vào***

**Input**

Gồm các số nguyên không quá 18 chữ số, không quá 100000 số

**Output**

In ra số nguyên lớn nhất trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 12 23 35 47 78 5908 102  3 4 5 76 89 960  346 227 1111 | 5908 |

**BÀI 2. Ctest002. TÌM TỪ DÀI NHẤT**

Cho một đoạn văn bản bất kỳ gồm vô hạn dòng. Hãy in ra từ có độ dài lớn nhất. Nếu có nhiều từ cùng độ dài lớn nhất thì in ra từ đầu tiên xuất hiện trong văn bản xét từ trái qua phải, trên xuống dưới.

***Sinh viên cần tự đọc hết luồng vào***

**Input**

Gồm một đoạn văn bản bất kỳ, nhiều dòng nhưng không quá 100000 từ

**Output**

In ra từ có độ dài lớn nhất kèm theo độ dài của nó

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| Ngon ngu lap trinh C++  Cau truc du lieu va giai do | trinh - 5 |

**BÀI 3. Ctest031. CHỮ SỐ LỚN NHẤT VÀ NHỎ NHẤT**

Cho số nguyên dương N có thể lên đến 500 chữ số. Hãy in ra chữ số nhỏ nhất, chữ số lớn nhất của số đó. Trường hợp tất cả các số bằng nhau thì chữ số nhỏ nhất và chữ số lớn nhất là chính số đó

**Input**

Dòng đầu tiên là số bộ test T

Mỗi bộ test là một số nguyên dương N cho trước

**Output**

Với mỗi bộ test, in ra chữ số lớn nhất, rồi đến chữ số nhỏ nhất. Hai số này cách nhau một khoảng trống

Kết quả của mỗi bộ test được in trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  1234  5555555555555555555555 | 1 4  5 5 |

**BÀI 4. Ctest016. LIỆT KÊ SỐ FIBONACCI**

Dãy số Fibonacci được xác định bằng công thức như sau:

F[0] = 0, F[1] = 1;

F[n] = F[n-1] + F[n-2] với mọi n >= 2.

Hãy liệt kê các số Fibonacci F[i], với i trong đoạn [a; b]

**Input**

Dòng đầu tiên là số bộ test T

Mỗi bộ test gồm 2 số a và b (a < b). Hai số cách nhau vài khoảng trống

**Output**

Với mỗi bộ test, in ra các số F[i] với a <=i<=b theo yêu cầu. Mỗi số cách nhau một khoảng trống

Kết quả của mỗi bộ test được in trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  1 10 | 1 1 2 3 5 8 13 21 34 55 |

**BÀI 5. Ctest036. TĂNG GIẢM 1 ĐƠN VỊ**

Chúng ta đã biết, trong ngôn ngữ lập trình C++ thì với một biến x bất kỳ, x++ hay ++x thì giá trị của x sẽ tăng thêm 1 đơn vị. Ngược lại, x--, --x thì giá trị của x sẽ giảm đi 1 đơn vị

Trong bài toán này, ta sẽ xét biến chỉ có 1 chữ cái. Biết rằng ban đầu biến có giá trị là 0 và biến nhận giá trị nguyên. Hỏi rằng sau các câu lệnh tăng giảm thì giá trị của biến là bao nhiêu?

**Input**

Dòng đầu tiên là số N là số câu lệnh

N dòng tiếp theo, mỗi dòng là một câu lệnh tăng giảm

**Output**

In ra giá trị sau cùng của biến

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  X++  --X | 0 |

**BÀI 6. DSA\_P007. DÃY CON LIÊN TIẾP**

Cho dãy số A[] có N phần tử, N không quá 105, các số trong dãy đều nguyên dương và không quá 9 chữ số. Hãy tính độ dài của dãy con liên tiếp có trung bình cộng lớn nhất trong dãy. Trong trường hợp có nhiều dãy con liên tiếp đều có trung bình cộng lớn nhất thì dãy nào dài hơn sẽ được chọn.

**Input**

Dòng đầu ghi số N.

Dòng thứ 2 ghi N số của dãy A[]

**Output**

Ghi ra độ dài dãy con liên tiếp có trung bình cộng lớn nhất tìm được.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  5 1 6 7 2 | 1 |

**Giải thích:** Dãy con liên tiếp có trung bình cộng lớn nhất là A = {a[3]} = {7} với trung bình cộng là 7

**BÀI 7. DSA\_P041. SẮP XẾP CHẴN LẺ**

Cho dãy số A[] có n phần tử. Hãy sắp xếp các số chẵn trong dãy theo thứ tự tăng dần và các số lẻ theo thứ tự giảm dần. In ra dãy kết quả đã sắp xếp trong đó vị trí số chẵn và vị trí số lẻ không thay đổi so với dãy ban đầu.

**Input**

Dòng đầu ghi số n (1 < n ≤ 1000)

Các dòng tiếp theo ghi đủ n số của dãy A[], các số đều nguyên dương và không quá 1000.

**Output**

Ghi ra dãy kết quả đã sắp xếp trong đó các vị trí của số chẵn và số lẻ không thay đổi.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Ouput** |
| 10  1 2 3 4 5 6 7 7 9 6 | 9 2 7 4 7 6 5 3 1 6 |

**BÀI 8. DSA\_P034. TÍCH CHỮ SỐ**

Tìm số nguyên dương nhỏ nhất N và có tích các chữ số là p

**Input**

Dòng đầu ghi số bộ test T

T dòng tiếp theo mỗi dòng là một số p không quá 18 chữ số

**Output**

Mỗi bộ test in ra số nguyên dương N thoả mãn trên một dòng. Nếu không có số N thoả mãn in -1

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Ouput** |
| 2  1  10 | 1  25 |

**BÀI 9. Ctest080. BIẾN ĐỔI VỀ 1**

Cho số nguyên dương N. Mỗi bước thực hiện các phép biến đổi N theo quy tắc sau

* Nếu N chẵn thì N = N/2
* Nếu N lẻ thì N = N\*3 + 1

Hãy đếm xem có bao nhiêu giá trị xuất hiện cho đến khi N = 1. Tất nhiên nếu ban đầu N = 1 thì chỉ có một giá trị duy nhất.

Ví dụ: N = 3 thì sẽ có 8 giá trị xuất hiện lần lượt là: 3, 10, 5, 16, 8, 4, 2, 1

**Input**

Có nhiều test, mỗi test ghi trên một dòng số nguyên dương N  không quá 100.

Input kết thúc khi N = 0.

**Output**

Với mỗi test, ghi ra kết quả tính được trên một dòng.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  2  3  0 | 1  2  8 |

**BÀI 10. DSA\_P027. TÁCH NHÓM TỐI ƯU**

Cho dãy số A[] có N phần tử là các số nguyên dương. Với mỗi số nguyên K, hãy tính xem có thể tách dãy số A thành ít nhất bao nhiêu nhóm sao cho mỗi số trong nhóm đều có thể tìm được ít nhất một số khác **cùng nhóm** có chênh lệch **không vượt quá K.**

Ví dụ: A[] = {2, 6, 1, 7, 3, 4, 9}; K = 1 thì kết quả là 3 ứng với 3 nhóm {2,1,3,4}; {6,7};  {9}

**Input**

Dòng đầu ghi hai số N và K (0 <= K <= 105; 0 <= N <= 106).

Dòng thứ 2 ghi ra N số của dãy A[], các số nguyên dương và không quá 106.

**Output**

Ghi ra số nhóm ít nhất có thể.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 7 1  2 6 1 7 3 4 9 | 3 |

**BÀI 11. DSA\_P006. QUÂN VUA TRÊN BÀN CỜ**

Trên bàn cờ vua kích thước 8\*8 thì quân vua được phép di chuyển đến cả 8 ô liền kề theo cả đường dọc và đường chéo. Tất nhiên quân vua sẽ không thể di chuyển được ra ngoài bàn cờ.

Cho 2 ô trên bàn cờ gọi là ô bắt đầu và ô kết thúc. Hãy tính xem quân vua cần ít nhất bao nhiêu bước để di chuyển từ ô bắt đầu đến ô kết thúc.

**Input**

Có 2 cặp số nguyên s1,s2 và f1,f2 lần lượt và vị trí ô bắt đầu và ô kết thúc. Các vị trí đảm bảo nằm trong phạm vi bàn cờ.

**Output**

Số bước đi ít nhất của quân vua

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 3 1 6 | 3 |
| 5 5 5 6 | 1 |

**BÀI 12. DSA\_P009. TỔNG ƯỚC SỐ**

Cho hai số nguyên dương a,b. Hãy đếm xem trong đoạn [a,b] có bao nhiêu số có tổng các ước số (không tính chính nó) lớn hơn giá trị của nó.

Ví dụ: số 12 có tổng ước số là 1 + 2 + 3 + 4 + 6 = 16 > 12.

**Input**

Chỉ có hai số a và b (1 <= a <= b <= 106).

**Output**

Ghi ra số lượng các số thỏa mãn.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 50 | 9 |

**HẾT**