

Bài giải tham khảo

**TÍCH HAI MA TRẬN:**

package BAI\_TAP;

import java.util.\*;

class Matran{

private int n;

private int [][]a;

public Matran(int n){

this.n = n;

this.a = new int[n][n];

}

public int [][] getMatran(){

return this.a;

}

public void input(Scanner sc){

for(int i = 0;i<this.n;i++){

for(int j = 0;j<this.n;j++){

this.a[i][j] = sc.nextInt();

}

}

}

public int getKT(){

return this.n;

}

public void out(){

for(int i = 0;i<this.n;i++){

for(int j = 0;j<this.n;j++){

System.out.print(this.a[i][j] + " ");

}

System.out.println();

}

}

public Matran tich(int[][] b) {

Matran res = new Matran(this.n);

for (int i = 0; i < this.n; i++) {

for (int j = 0; j < this.n; j++) {

res.a[i][j] = 0;

for (int k = 0; k < this.n; k++) {

res.a[i][j] += this.a[i][k] \* b[k][j];

}

}

}

return res;

}

}

public class TICH\_HAI\_MA\_TRAN {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

//nhap kich thuoc

int n = sc.nextInt();

Matran m = new Matran(n);

//nhap ma tran a

m.input(sc);

Matran b = new Matran(m.getKT());

//nhap ma tran b

b.input(sc);

Matran t = m.tich(b.getMatran());

//viet ra ma tran tich

t.out();

}

}

**TẠO LỚP SẢN PHẨM**

package BAI\_TAP;

import java.util.\*;

class SanPham {

private static int counter = 0;

private String id;

private String ten, nhaSX;

private int soLuong;

private double gia, thanhTien;

public SanPham(String ten, String nhaSX, int soLuong, double gia) {

counter++;

this.id = String.format("%03d", counter);

this.ten = ten;

this.nhaSX = nhaSX.toUpperCase() + "-" + this.id;

this.soLuong = soLuong;

this.gia = gia;

this.thanhTien = soLuong \* gia;

if (this.soLuong >= 20) this.thanhTien = this.thanhTien \* 90 / 100;

}

@Override

public String toString() {

return this.nhaSX + " " + this.ten + " " + + this.soLuong + " " + String.format("%.1f %.1f", this.gia, this.thanhTien);

}

}

public class TAO\_LOP\_SAN\_PHAM {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = Integer.parseInt(sc.nextLine());

String ten;

String nhaSX;

int soLuong;

double gia;

while (n-- > 0) {

ten = sc.nextLine();

nhaSX = sc.nextLine();

soLuong = Integer.parseInt(sc.nextLine());

gia = Double.parseDouble(sc.nextLine());

SanPham p = new SanPham(ten, nhaSX, soLuong, gia);

System.out.println(p);

}

}

}

/\*

2

Galaxy A53

samsung

10

100

C21 Plus

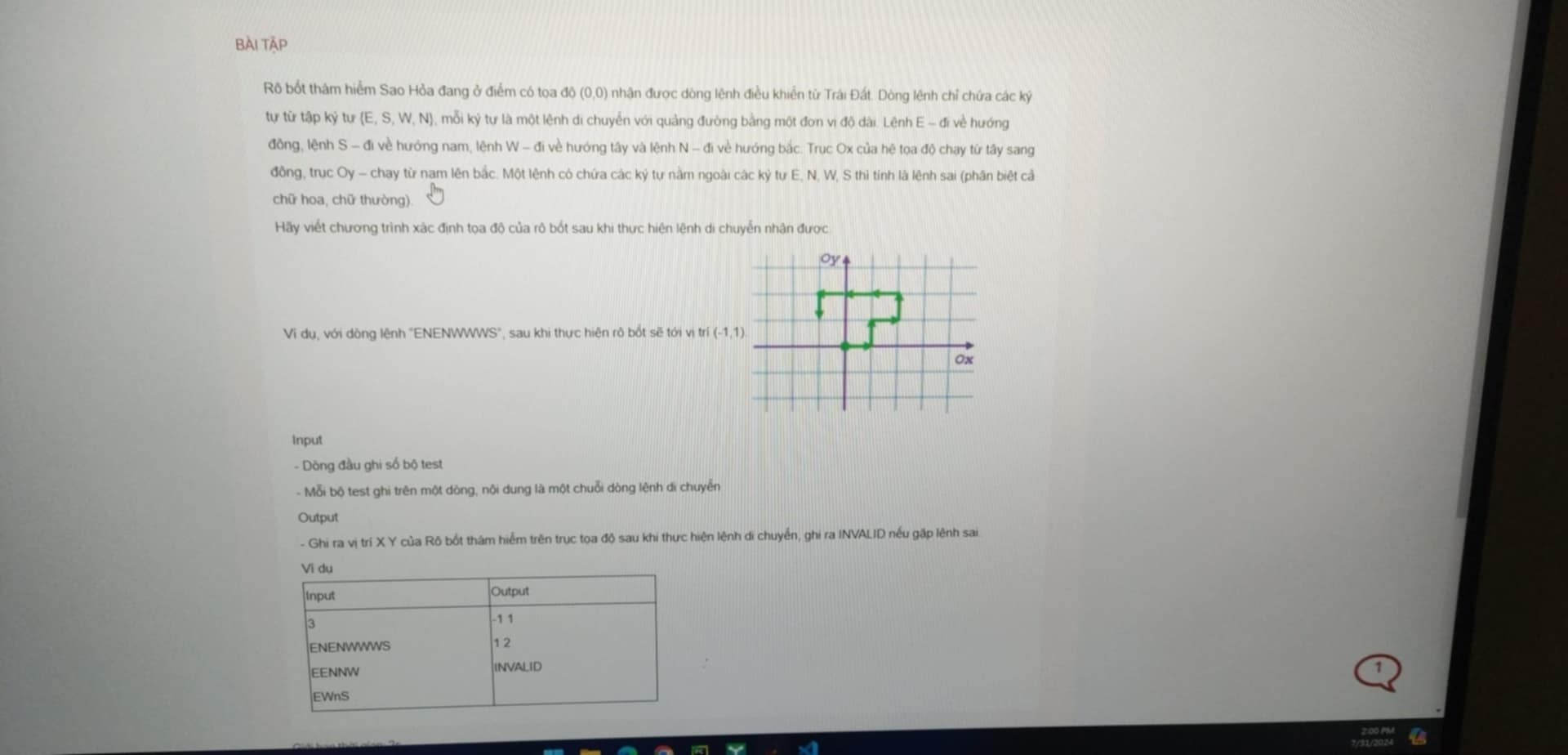
Nokia

30

100

\*/

**Java\_HA\_1. Rô bốt thám hiểm**



**BÀI KIỂM TRA SỐ 1 – LỚP D22\_N06 – ĐỀ A**

**BÀI 1. J99106. TỔNG VÀ TÍCH CÁC SỐ KHÁC NHAU**

Cho file văn bản DATA.in có không quá 5000 số nguyên dương, các số có không quá 9 chữ số. Hãy thực hiện tính tổng và tích các số **phân biệt** trong file (Tức là số nào xuất hiện 2 lần thì chỉ lấy 1 lần khi tính tổng hoặc tính tích. Chẳng hạn: 2 2 5 5. Tổng: 2 + 5 = 7. Tích: 2.5 = 10)

**Input:** File văn bản DATA.in như đã mô tả

**Output:** Gồm 2 dòng

* Dòng đầu ghi tổng của các số khác nhau
* Dòng thứ hai ghi tích của các số khác nhau

**Ví dụ**

|  |  |
| --- | --- |
| **DATA.in** | **Output** |
| 1 3 2 1 3 4 5 4 5 6  1 2 3 | 21  720 |

**BÀI 2. CHIA VÀ SẮP XẾP**

Cho mảng A gồm N số nguyên. Bạn hãy chia mảng A thành hai phần, lấy chỉ số M làm gốc (0 M < N)

Các phần tử từ đầu mảng đến ngay trước vị trí M (index M – 1) được sắp xếp tăng dần. Các phần tử từ vị trí M đến hết mảng được sắp xếp giảm dần

**Input**

Dòng đầu tiên là số bộ test. Mỗi bộ test gồm 2 dòng

* Dòng thứ nhất là 2 số N và M
* Dòng thứ hai là mảng A gồm N số nguyên

**Output**

In ra hai phần của dãy số A đã được sắp xếp theo yêu cầu đề bài. Hai dãy số ở trên cùng một dòng, mỗi số cách nhau đúng 1 dấu cách. Mỗi bộ test ghi kết quả trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  6 3  10 1 5 6 2 4 | 1 5 10 6 4 2 |

**BÀI 3. BIẾN ĐỔI CHO TỔNG LỚN NHẤT**

Cho một mảng a gồm n phần tử và 1 số k, bạn phải thực hiện đúng k phép biến đổi, với mỗi phép biến đổi, bạn phải chọn 1 số a[i] và thay thế nó bằng số -a[i].

Hãy tìm cách thực hiện k phép biến đổi sao cho tổng các phần tử của mảng a sau khi biến đổi là lớn nhất.

**Input**

Dòng đầu tiên là số n (1 ≤ n ≤ 105) và k (0 ≤ k ≤ 105)

Dòng tiếp theo gồm n số a[1], a[2], ... a[n] ( -105 ≤ a[i] ≤ 105)

**Output:**

1 dòng duy nhất là tổng lớn nhất của các phần tử của mảng a sau khi thực hiện phép biến đổi.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 1  4 6 2 | 8 |
| 5 2  -3 10 6 1 2 | 20 |
| 5 2  3 10 6 1 2 | 22 |

Giải thích test:

* Test 1: Ta sử dụng phép biến đổi với phần tử a[3] = 2. Khi đó mảng trở thành [4, 6, -2] , tổng = 8
* Test 2: Cần đổi dấu 2 lần. Ta chọn -3 và 1. Khi đó, mảng trở thành [3, 10, 6, -1, 2]. Tổng các phần tử lúc này là 20
* Test 3: Cần đổi dấu 2 lần. Có thể chọn bất kỳ số nào trong dãy để đổi dấu hai lần. Khi ấy, mảng vẫn giữ nguyên trạng thái ban đầu, tổng là 22.

**BÀI KIỂM TRA SỐ 1 – LỚP D22\_N06 – ĐỀ B**

**BÀI 1. TỔNG VÀ TÍCH**

Cho file văn bản DATA.in có không quá 5000 số nguyên dương, các số có không quá 9 chữ số. Hãy thực hiện tính tổng và tích các số trong file

**Input:** File văn bản DATA.in như đã mô tả

**Output:** Gồm 2 dòng

* Dòng đầu ghi tổng của các số khác nhau
* Dòng thứ hai ghi tích của các số khác nhau

**Ví dụ**

|  |  |
| --- | --- |
| **DATA.in** | **Output** |
| 1 2 3 4 5 6  1 2 3 | 27  4320 |

**BÀI 2. CHIA VÀ SẮP XẾP 2**

Cho mảng A gồm N số nguyên và 2 số K, M. Bạn hãy chèn số K vào ngay trước A[M], rồi chia mảng A thành hai phần, lấy chỉ số M làm gốc (0 M < N)

Các phần tử từ đầu mảng đến ngay trước vị trí M (index M – 1) được sắp xếp tăng dần. Các phần tử từ vị trí M đến hết mảng được sắp xếp giảm dần

**Input**

Dòng đầu tiên là số bộ test. Mỗi bộ test gồm 2 dòng

* Dòng thứ nhất là 3 số N, M và K
* Dòng thứ hai là mảng A gồm N số nguyên

**Output**

In ra hai phần của dãy số A đã được sắp xếp theo yêu cầu đề bài. Hai dãy số ở trên cùng một dòng, mỗi số cách nhau đúng 1 dấu cách. Mỗi bộ test ghi kết quả trên một dòng

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  6 3 0  10 1 5 6 2 4 | 0 1 5 10 6 4 2 |

**MỘT SỐ CÂU THỰC HÀNH Ngày 06/10/2024**

**BÀI 1. JP013. LIỆT KÊ CÁC SỐ CÓ HAI CHỮ SỐ TĂNG DẦN**

Cho một dãy ký tự số không quá 1000 chữ số và không có chữ số 0.

Người ta lần lượt lấy ra mỗi lần 2 chữ số tính từ trái sang phải. Nếu bước cuối cùng không đủ hai chữ số thì bỏ qua chữ số đó. Kết quả sẽ được một dãy số nguyên dương A[] chỉ bao gồm các số có hai chữ số.

Hãy liệt kê **các số khác nhau xuất hiện trong A[]** theo **thứ tự tăng dần**.

**Input**

Chỉ có một dòng ghi dãy ký tự số (độ dài không quá 1000). Dữ liệu vào đảm bảo không có chữ số 0.

**Output**

Ghi ra lần lượt các số khác nhau xuất hiện trong dãy A[] theo thứ tự tăng dần, mỗi số viết cách nhau một khoảng trống.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 124356141111434356149 | 11 12 14 43 56 |

**BÀI 2. JP015. TÁCH SỐ VÀ SẮP XẾP**

Cho N xâu ký tự bao gồm cả chữ số và chữ cái. Các chữ số liên tiếp sẽ tạo ra một số nguyên. Hãy sắp xếp các số tách được theo thứ tự tăng dần.

Chú ý: các chữ số 0 ở đầu nếu có sẽ không được tính. Ví dụ: các chữ số tách ra được là 00234 thì được tính như số 234, nếu là 00000000 thì được tính như số 0.

**Input**

Dòng đầu ghi số N (không quá 100).

N dòng tiếp theo, mỗi dòng ghi một xâu ký tự, độ dài không quá 100.

Dữ liệu đảm bảo sẽ tách ra được không quá 500 số.

**Output**

Ghi ra các số theo thứ tự sắp xếp tăng dần, mỗi số trên một dòng.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  A129h  G07bxjq3  aaaaaaa4aaaa | 3  4  7  129 |

**BÀI 3. JP017. ĐÁNH DẤU SẢN PHẨM LỖI**

Sau khi qua công đoạn đóng gói, các sản phẩm của công ty X sẽ được đưa qua máy phát hiện lỗi. Các sản phẩm lỗi sẽ được trả lại và thứ tự của sản phẩm đó sẽ được ghi vào file log. Dựa trên file log của hệ thống, hãy liệt kê danh sách chi tiết các sản phẩm bị lỗi và các sản phẩm tốt

**Input**: Nhập từ bàn phím

* Dòng đầu ghi 2 số N, M là tổng số sản phẩm và số sản phẩm bị lỗi

(2 ≤ N ≤ 100000; 1 ≤ m ≤ min (N, 1000)

* Tiếp theo là M giá trị số, ghi rõ thứ tự các sản phẩm lỗi theo thứ tự tăng dần. Các giá trị này đảm bảo trong phạm vi từ 1 đến N

**Output:** In ra màn hình 2 dòng:

* Dòng 1 là các sản phẩm đơn hoặc đoạn liên tiếp các sản phẩm bị lỗi theo thư tự tăng dần
* Dòng 1 là các sản phẩm đơn hoặc đoạn liên tiếp các sản phẩm viết chính xác theo thư tự tăng dần

*Chú ý: Sử dụng dấu gạch ngang để thể hiện đoạn liên tiếp. Xem ví dụ để hiểu rõ hơn cách ghi kết quả*

**Ví dụ**

|  |
| --- |
| **Input** |
| 100 10  2 3 5 10 11 12 25 26 88 89 |
| **Output** |
| Errors: 2-3 5 10-12 25-26 and 88-89  Not Errors: 1 4 6-9 13-24 27-87 and 90-100 |

**BÀI 4. JKT016. KIỂM TRA CÂU VIẾT ĐÚNG QUY TẮC**