

[OOP]. Bài 1. Lớp phân số

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Phân số gồm hai thuộc tính private là tử số và mẫu số. Sau đó thực hiện nhập vào một phân số và in ra phân số đó ở dạng tối giản.

Input Format

Có hai số nguyên dương lần lượt là tử số và mẫu số.

Constraints

Các giá trị đều nguyên dương và không quá 18 chữ số.

Output Format

Ghi ra phân số tối giản;

Sample Input 0

334 37

Sample Output 0

334/37

Sample Input 1

312 945

Sample Output 1

104/315

[OOP]. Bài 2. Lớp sinh viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Thí Sinh gồm các thông tin: Họ tên, Ngày sinh, Điểm môn 1, Điểm môn 2, Điểm môn 3 và Tổng điểm. Đọc thông tin 1 thí sinh từ bàn phím và in ra màn hình 3 thông tin: Họ tên, Ngày sinh, Tổng điểm.

Input Format

Gồm 5 dòng lần lượt, mỗi dòng ghi 1 thông tin: Họ tên, Ngày sinh, Điểm môn 1, Điểm môn 2, Điểm môn 3.

Constraints

Các giá trị điểm là số thực (float). Họ tên không quá 50 chữ cái. Ngày sinh viết đúng chuẩn dd/mm/yyyy.

Output Format

Ghi ra Họ tên, Ngày sinh và Tổng điểm. Mỗi thông tin cách nhau một khoảng trống. Điểm được ghi ra với 1 số sau dấu phẩy.

Sample Input 0

```
Hoang Dinh Nam
28/09/2000
6.1
6.9
6.4
```

Sample Output 0

```
Hoang Dinh Nam 28/09/2000 19.4
```

[OOP]. Bài 3. Lớp Sinh Viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Sinh Viên gồm các thông tin: Mã SV, Họ tên, Lớp, Ngày sinh và Điểm GPA (dạng số thực float). Hàm khởi tạo không có tham số, gán các giá trị thuộc tính ở trạng thái mặc định (xâu ký tự rỗng, giá trị số bằng 0). Đọc thông tin 1 sinh viên từ bàn phím (không có mã sinh viên) và in ra màn hình. Trong đó Mã SV được gán là SV001. Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy.

Input Format

Gồm 4 dòng : 1. Họ tên không quá 30 chữ cái.

2. Lớp là một chuỗi không có dấu cách
3. Ngày sinh có đủ 3 phần ngày tháng năm nhưng có thể chưa đúng chuẩn dd/mm/yyyy.
4. Điểm GPA đảm bảo trong thang điểm 4 với 2 nhiều nhất 2 số sau dấu phẩy.

Constraints

N/A

Output Format

Ghi thông tin sinh viên trên 1 dòng, mỗi thông tin cách nhau 1 khoảng trống, gpa lấy 1 chữ số sau dấu phẩy

Sample Input 0

```
Nguyen Van Nam  
CNTT1  
16/8/2001  
2.8
```

Sample Output 0

```
SV001 Nguyen Van Nam CNTT1 16/08/2001 2.8
```

[OOP]. Bài 4. Lớp nhân viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Một nhân viên làm việc trong công ty được lưu lại các thông tin sau:

- 1. Mã nhân viên: được gán giá trị là 00001
- 2. Họ tên: Xâu ký tự không quá 50 chữ cái.
- 3. Giới tính: Nam hoặc Nu
-

4. Ngày sinh: đúng theo chuẩn dd/mm/yyyy

-

5. Địa chỉ: Xâu ký tự không quá 100 chữ cái

-

6. Mã số thuế: Dãy số có đúng 10 chữ số

-

7. Ngày ký hợp đồng: đúng theo chuẩn dd/mm/yyyy

Viết chương trình nhập một nhân viên (không nhập mã) in ra màn hình thông tin của nhân viên đó.

Input Format

Gồm 6 dòng lần lượt ghi các thông tin theo thứ tự đã ghi trong đề bài. Không có mã nhân viên.

Constraints

N/A

Output Format

Ghi ra đầy đủ thông tin nhân viên trên một dòng, các thông tin cách nhau đúng một khoảng trống.

Sample Input 0

```
Nguyen Thi Nhung
Nu
28/09/1999
Ha Nam
8850288467
13/07/2021
```

Sample Output 0

```
00001 Nguyen Thi Nhung Nu 28/09/1999 Ha Nam 8850288467 13/07/2021
```

[OOP]. Bài 5. Lớp Nhân Viên 2

-

[Problem](#)

-

[Submissions](#)

- [Discussions](#)

Một nhân viên làm việc trong công ty được lưu lại các thông tin sau: 1. Mã nhân viên: được gán giá trị là 00001. 2. Họ tên: Xâu ký tự không quá 50 chữ cái. 3. Giới tính: Nam hoặc Nu. 4. Ngày sinh, có thể chưa chuẩn nhưng đủ 3 phần và phân cách bởi dấu /. 5. Địa chỉ: Xâu ký tự không quá 100 chữ cái. 6. Mã số thuế: Dãy số có đúng 10 chữ số. 7. Ngày ký hợp đồng, có thể chưa chuẩn nhưng đủ 3 phần và phân cách bởi dấu /. Viết chương trình nhập một nhân viên (không nhập mã) in ra màn hình thông tin của nhân viên đó. Tên nhân viên được chuẩn hóa về dạng viết hoa chữ cái đầu của từng từ, ngày sinh về đúng dạng chuẩn dd/mm/yyyy

Input Format

Gồm 6 dòng lần lượt ghi các thông tin theo thứ tự đã ghi trong đề bài.

Constraints

N/A

Output Format

Ghi ra đầy đủ thông tin nhân viên trên một dòng, các thông tin cách nhau đúng một khoảng trống.

Sample Input 0

Hoang Dinh NAM
Nam
9/9/1998
Bac Ninh
8743766435
10/2/2021

Sample Output 0

00001 Hoang Dinh Nam Nam 09/09/1998 Bac Ninh 8743766435 10/02/2021

[OOP]. Bài 6. Lớp HCN

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

[illegible]

Rectangle
- width: double - height: double - <u>color: String</u>
+ Rectangle() + Rectangle(width: double, <div style="padding-left: 100px;">height: double, color: String)</div> + getWidth():double + setWidth(width:double): void + getHeight(): double + setHeight(height:double): void + <u>getColor(): String</u> + <u>setColor(color): void</u> + findArea(): double + findPerimeter(): double

Viết chương trình

nhập vào giá trị độ dài hai cạnh của hình chữ nhật và màu sắc. In ra thông tin về chu vi, diện tích và màu sắc (đã đưa về dạng chuẩn trong đó ký tự đầu viết hoa, các ký tự sau viết thường) của hình chữ nhật đó.

Input Format

Gồm 2 số nguyên a, b là độ dài 2 cạnh hình chữ nhật và một xâu ký tự (không có khoảng trống) mô tả màu sắc.

Constraints

$$-1000 \leq a, b \leq 1000;$$

Output Format

Nếu hình chữ nhật là hợp lệ (các cạnh đều nguyên dương) thì in ra 3 thông tin: chu vi, diện tích, màu sắc, mỗi thông tin cách nhau một khoảng trống. Nếu dữ liệu không hợp lệ in ra INVALID

Sample Input 0

22 25 Gray

Sample Output 0

94 550 Gray

[OOP]. Bài 9. Thu nhập của giáo viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trường phổ thông XYZ tính lương giáo viên theo quy tắc sau: Mỗi giáo viên có mã ngạch gồm 4 ký tự trong đó. 2 ký tự đầu là chức vụ (HT: Giáo viên kiêm nhiệm Hiệu trưởng, HP: Giáo viên kiêm nhiệm Hiệu phó, GV: Giáo viên thường). 2 ký tự số cuối cùng cho biết hệ số bậc lương (không quá 20). Lương cơ bản của mỗi giáo viên cũng có thể khác nhau. Phụ cấp quy định như sau. HT: 2.000.000. HP: 900.000. GV: 500.000. Thu nhập được tính bằng lương cơ bản nhân với hệ số bậc lương cộng thêm phụ cấp. Hãy tính lương cho 1 giáo viên theo quy tắc trên.

Input Format

Có 3 dòng lần lượt là mã ngạch, họ tên và lương cơ bản.

Constraints

Lương cơ bản là số nguyên không quá 10^7

Output Format

Chỉ có một dòng ghi lần lượt các thông tin: mã ngạch, họ tên, bậc lương, thu nhập.

Sample Input 0

HP05
Nguyen Van Ha
1748000

Sample Output 0

HP05 Nguyen Van Ha 5 9640000

[OOP]. Bài 11. Tính lương nhân viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Thông tin về nhân viên bao gồm: Mã nhân viên (tự động tăng theo thứ tự nhập, tính từ NV01, nếu chỉ có 1 nhân viên thì mã là NV01) - Họ và tên - Lương cơ bản mỗi ngày công - Số ngày công - Chức vụ Tiền lương được tính bằng lương cơ bản nhân với số ngày công. Giả sử quy tắc tính tiền thưởng như sau:

- Số ngày công từ 25 trở lên thì thưởng 20% tiền lương
- Số ngày công từ 22 đến dưới 25 thì thưởng 10% tiền lương
- Dưới 22 ngày công thì không có thưởng. Mỗi nhân viên có thể có thêm phụ cấp chức vụ:
 - GD: 250000
 - PGD: 200000
 - TP: 180000
- NV: 150000 Hãy nhập thông tin 1 nhân viên và tính toán thu nhập theo quy tắc trên.

Input Format

Gồm 4 dòng lần lượt ghi Họ tên, lương cơ bản, số ngày công và chức vụ; Số ngày công là số nguyên không âm không vượt quá 31;

Constraints

Lương cơ bản là số nguyên dương không quá 10^7 ;

Output Format

Ghi ra một dòng gồm: mã nhân viên, tên nhân viên, lương tháng, thưởng, phụ cấp và thu nhập. Mỗi thông tin cách nhau một khoảng trống.

Sample Input 0

```
Nguyen Van Teo
26000
```


Sample Output 0

NV01 Nguyen Van Teo 546000 0 200000 746000

[OOP]. Bài 12. Tuyển sinh

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trường Đại học XYZ tuyển sinh theo hình thức xét điểm thi ba môn Toán – Lý – Hóa. Để đơn giản, khu vực tuyển sinh được quy định luôn bởi ba chữ cái đầu tiên trong mã thí sinh. Do rất thích các thí sinh đến từ Khu vực 3 nên trường XYZ tự quy định giá trị điểm ưu tiên Khu vực như trong bảng sau:

Bảng Ưu Tiên Khu Vực	
Mã khu vực	Điểm Ưu Tiên
KV1	0.5
KV2	1.0
KV3	2.5

Giả sử biết trước điểm chuẩn là 24.

Hãy xác định tình trạng tuyển sinh của thí sinh.

Input Format

Chỉ bao gồm thông tin của một thí sinh trên 5 dòng lần lượt là: - Mã thí sinh - Họ tên - Điểm toán - Điểm lý - Điểm hóa

Constraints

Các giá trị điểm đều đảm bảo trong phạm vi $[0,10]$ và có thể có 1 chữ số phần thập phân.

Output Format

Ghi ra các thông tin: - Mã thí sinh - Họ tên - Khu vực - Tổng điểm – có tính ưu tiên (có thể có 1 số phần thập phân), trong trường hợp điểm là số thực với phần thập phân bằng 0

thì không in ra phần thập phân. - Trạng thái: TRUNG TUYEN hoặc TRUOT (sau khi đã tính cả điểm ưu tiên)

Sample Input 0

```
KV2B
Hoang Ngoc Long
3.4
9
0
```

Sample Output 0

```
KV2B Hoang Ngoc Long 2 13.4 TRUOT
```

[OOP]. Bài 13. Lớp phân số

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Phân số là sự biểu diễn số hữu tỷ dưới dạng tỷ lệ của hai số nguyên, trong đó số ở trên được gọi là tử số, còn số ở dưới được gọi là mẫu số. Cho hai phân số A và B có tử số và mẫu số được nhập từ bàn phím. Viết chương trình thực hiện hai nhiệm vụ sau: 1. Thực hiện phép tính $C = (A + B)^2$ và rút gọn kết quả. 2. Thực hiện phép tính $D = A \times B \times C$ và rút gọn kết quả.

Input Format

Dòng đầu tiên là số bộ test T; T dòng tiếp theo, mỗi dòng gồm 4 số lần lượt là tử và mẫu số của phân số A và phân số B

Constraints

($T \leq 100$); Các phân số có tử số và mẫu số là số nguyên trị tuyệt đối ≤ 100 ; Dữ liệu đảm bảo mẫu số khác 0.

Output Format

Ghi ra kết quả của hai phép tính theo định dạng phân số, cách nhau một khoảng trống.

Sample Input 0

```
6
1 2 3 4
2 3 4 5
-23 -36 99 90
```

24 -1 99 60
-40 -9 98 91
37 -31 61 92

Sample Output 0

25/16 75/128
484/225 3872/3375
97969/32400 24786157/11664000
199809/400 19781091/-1000
417316/13689 233696960/1601613
2289169/8133904 5166654433/-23197894208

[OOP]. Bài 15. Lớp Set 1

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trong lý thuyết tập hợp, một tập hợp chỉ được phép chứa các giá trị phân biệt và luôn luôn lưu các giá trị theo thứ tự tăng dần. Khai báo lớp `IntSet` và viết các phương thức để thực hiện các thao tác trên tập hợp số nguyên. Sử dụng lớp `IntSet` để in ra tập hợp các số nguyên là giao của hai tập số trong 2 dãy ban đầu. Sử dụng hàm `main` có nội dung như sau :

```
public static void main(String[] args){  
    Scanner sc = new Scanner(System.in);  
    int n = sc.nextInt(), m = sc.nextInt(), a[] = new int[n], b[] = new int[m];  
    for(int i = 0; i<n; i++) a[i] = sc.nextInt();  
    for(int i = 0; i<m; i++) b[i] = sc.nextInt();  
    IntSet s1 = new IntSet(a);  
    IntSet s2 = new IntSet(b);  
    IntSet s3 = s1.intersection(s2);  
    System.out.println(s3);  
}
```

Input Format

Dòng đầu ghi 2 số n và m ($1 < n, m < 100$). Dòng thứ 2 ghi n số của $a[]$. Dòng thứ 3 ghi m số của $b[]$. Các số đều dương và nhỏ hơn 1000.

Constraints

N/A

Output Format

Ghi ra hợp của hai tập theo thứ tự tăng dần.

Sample Input 0

```
14 5
92 36 73 11 78 19 57 76 96 58 36 38 29 42
19 21 15 4 78
```

Sample Output 0

```
19 78
```

[OOP]. Bài 16. Lớp Set 2

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trong lý thuyết tập hợp, một tập hợp chỉ được phép chứa các giá trị phân biệt và luôn luôn lưu các giá trị theo thứ tự tăng dần. Khai báo lớp IntSet và viết các phương thức để thực hiện các thao tác trên tập hợp số nguyên. Sử dụng lớp IntSet để in ra tập hợp các số nguyên là hợp của hai tập số trong 2 dãy ban đầu. Yêu cầu sử dụng hàm main có nội dung như sau để chạy chương trình.

```
public static void main(String[] args){
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt(), m = sc.nextInt(), a[] = new int[n], b[] = new int[m];
    for(int i = 0; i<n; i++) a[i] = sc.nextInt();
    for(int i = 0; i<m; i++) b[i] = sc.nextInt();
    IntSet s1 = new IntSet(a);
    IntSet s2 = new IntSet(b);
    IntSet s3 = s1.union(s2);
    System.out.println(s3);
}
```

Input Format

Dòng đầu ghi 2 số n và m ($1 < n, m < 100$). Dòng thứ 2 ghi n số của a[]. Dòng thứ 3 ghi m số của b[]. Các số đều dương và nhỏ hơn 1000;

Constraints

N/A

Output Format

Ghi ra hợp của hai tập theo thứ tự tăng dần.

Sample Input 0

12 5
89 16 18 57 93 25 40 25 75 71 30 18
23 93 8 16 80

Sample Output 0

8 16 18 23 25 30 40 57 71 75 80 89 93

[OOP]. Bài 17. Lớp set 3

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trong lập trình cơ bản, một từ được hiểu là một dãy ký tự liên tiếp không chứa khoảng trống, dấu tab hoặc dấu xuống dòng. Xây dựng lớp WordSet để quản lý tập hợp các từ khác nhau trong một xâu ký tự, sau khi đã chuyển hết về dạng chữ thường. Khi liệt kê thì tập từ thì sẽ luôn theo thứ tự từ điển tăng dần. Viết chương trình nhập hai dòng ký tự và hiển thị hợp và giao của hai tập từ tương ứng. Sử dụng hàm main có nội dung sẵn như sau :

```
1 public static void main(String[] args) {  
2     Scanner sc = new Scanner(System.in);  
3     WordSet s1 = new WordSet(sc.nextLine());  
4     WordSet s2 = new WordSet(sc.nextLine());  
5     System.out.println(s1.union(s2));  
6     System.out.println(s1.intersection(s2));  
7 }  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100
```

Input Format

Chỉ có 2 dòng, mỗi dòng có độ dài không quá 1000 ký tự.

Constraints

N/A

Output Format

Dòng 1 ghi hợp của 2 tập từ theo thứ tự từ điển. Dòng 2 ghi giao của 2 tập từ theo thứ tự từ điển.

Sample Input 0

```
aabc cbcac bbaa aaa acaba ccc abaaa  
caa aaa abaa aba abc acb aacb aabaa
```

Sample Output 0

```
aaa aabaa aabc aacb aba abaa abaaa abc acaba acb bbaa caa cbcac ccc  
aaa
```

[OOP]. Bài 18. Lớp ma trận

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Matrix mô tả ma trận các số nguyên. Sau đó nhập và tính tích hai ma trận A cỡ $n \times m$ và ma trận B cỡ $m \times p$. Với $1 < n, m, p < 50$. Các giá trị trong ma trận đều nguyên dương và không vượt quá 1000. Sử dụng hàm main có sẵn :

```
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
    int n = sc.nextInt(), m = sc.nextInt(), p = sc.nextInt();  
    Matrix a = new Matrix(n,m);  
    a.nextMatrix(sc);  
    Matrix b = new Matrix(m,p);  
    b.nextMatrix(sc);  
    System.out.println(a.mul(b));  
}
```

Input Format

Dòng đầu ghi 3 số n, m, p ; n dòng tiếp theo ghi ma trận A; m dòng tiếp theo ghi ma trận B;

Constraints

N/A

Output Format

In ra tích của ma trận

Sample Input 0

```
3 2 4  
147 790
```

393 462
675 445
690 172 776 550
344 761 900 713

Sample Output 0

373190 626474 825072 644120
430098 419178 720768 545556
618830 454745 924300 688535

[OOP]. Bài 19. Lớp ma trận 2

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Khai báo lớp Matrix mô tả ma trận các số nguyên. Nhập ma trận A cấp $N * M$. Hãy viết chương trình tính tích của A với ma trận chuyển vị của A.

Sử dụng hàm main có nội dung sẵn như sau :

```
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
    int n = sc.nextInt(), m = sc.nextInt();  
    Matrix a = new Matrix(n,m);  
    a.nextMatrix(sc);  
    Matrix b = a.trans();  
    System.out.println(a.mul(b));  
}
```

Input Format

Dòng đầu tiên ghi hai số n và m là bậc của ma trận a; n dòng tiếp theo, mỗi dòng ghi m số của một dòng trong ma trận A.

Constraints

n và m đều nguyên dương và nhỏ hơn 50. Các giá trị trong ma trận không vượt quá 100.

Output Format

Với mỗi bộ test ghi ra ma trận tích tương ứng, mỗi số cách nhau đúng một khoảng trống.

Sample Input 0

```
5 2
198 725
686 514
183 392
392 717
134 62
```

Sample Output 0

```
564829 508478 320434 597441 71482
508478 734792 327026 637450 123792
320434 327026 187153 352800 48826
597441 637450 352800 667753 96982
71482 123792 48826 96982 21800
```

[OOP]. Bài 20. Lớp Time

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Giá trị độ đo thời gian được biểu diễn bởi ba thành phần: giờ, phút, giây. Cho N giá trị thời gian, hãy sắp xếp danh sách theo thứ tự tăng dần.

Input Format

Dòng đầu ghi số nguyên dương N (không quá 5000) là số lượng giá trị thời gian cần sắp xếp. Mỗi giá trị thời gian biểu diễn trên một dòng bằng ba số nguyên dương, lần lượt là số giờ, số phút, số giây. Trong đó số giờ đảm bảo nhỏ hơn 100, số phút và số giây đảm bảo đúng quy tắc (tức là không quá 59).

Constraints

N/A

Output Format

In ra danh sách đã sắp xếp theo thứ tự tăng dần.

Sample Input 0

```
5
25 2 33
18 3 17
12 6 19
1 6 12
```


Sample Output 0

```
1 6 12
12 6 19
18 3 17
25 2 33
98 51 26
```

[OOP]. Bài 21. Danh sách mặt hàng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Bài toán quản lý danh sách mặt hàng trong đó mỗi mặt hàng sẽ có các thông tin: - Mã mặt hàng: tự động tăng, tính từ MH0001.

- Tên mặt hàng: chuỗi ký tự độ dài không quá 100.
- Đơn vị tính: chuỗi ký tự độ dài không quá 10.
- Giá mua: số nguyên dương không quá 7 chữ số.
- Giá bán: số nguyên dương không quá 7 chữ số.

Viết chương trình nhập danh sách mặt hàng, sắp xếp theo lợi nhuận (giá bán trừ đi giá mua) giảm dần. Nếu lợi nhuận bằng nhau thì in ra theo thứ tự mã tăng dần.

Input Format

Dòng đầu ghi số M là số mặt hàng (không quá 5000). Tiếp theo là thông tin của M mặt hàng, mỗi mặt hàng ghi trên 4 dòng theo đúng thứ tự đã mô tả (không có mã)

Constraints

N/A

Output Format

Ghi ra danh sách mặt hàng có đầy đủ thông tin ở trên và lợi nhuận tính được (mỗi thông tin cách nhau một khoảng trống)

Sample Input 0

Ca phe
Thung
388000
492000
Dep
Thung
376000
495000
Giay Nguoi Lon
Cai
389000
580000
Giay Nguoi Lon
Thung
383000
503000
Dep
Thung
484000
647000
Vay Nu
Cai
366000
516000

Sample Output 0

MH0003 Giay Nguoi Lon Cai 389000 580000 191000
MH0005 Dep Thung 484000 647000 163000
MH0006 Vay Nu Cai 366000 516000 150000
MH0004 Giay Nguoi Lon Thung 383000 503000 120000
MH0002 Dep Thung 376000 495000 119000
MH0001 Ca phe Thung 388000 492000 104000

[OOP]. Bài 22. Sinh viên 1

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Sinh Viên gồm các thông tin: Mã SV, Họ tên, Ngày sinh, Lớp và Điểm GPA (dạng số thực). Hàm khởi tạo không có tham số, gán các giá trị thuộc tính ở trạng thái mặc định (xâu ký tự rỗng, giá trị số bằng 0). Đọc thông tin N sinh viên từ bàn phím (không có mã sinh viên) và in ra lần lượt màn hình mỗi dòng 1 sinh viên theo đúng thứ tự ban đầu. Trong đó Mã SV được tự tạo ra theo quy tắc thêm mã SV sau đó là giá trị nguyên tự động tăng tính từ 001 (tối đa là 099). Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy

Input Format

Dòng đầu tiên ghi số sinh viên N ($0 < N < 50$). Mỗi sinh viên ghi trên 4 dòng lần lượt là Họ tên, Lớp, Ngày sinh và Điểm GPA. Trong đó: • Họ tên không quá 30 chữ cái. • Lớp là một chuỗi không có dấu cách. • Ngày sinh có đủ 3 phần ngày tháng năm nhưng có thể chưa đúng chuẩn dd/mm/yyyy. • Điểm GPA đảm bảo trong thang điểm 4 với 2 nhiều nhất 2 số sau dấu phẩy.

Constraints

N/A

Output Format

Ghi ra danh sách lần lượt các sinh viên có đầy đủ Mã sinh viên, Họ tên, Lớp, Ngày sinh (đã chuẩn hóa về dạng dd/mm/yyyy), Điểm GPA (với đúng 2 số sau dấu phẩy). Mỗi sinh viên ghi trên 1 dòng, mỗi thông tin cách nhau 1 khoảng trống.

Sample Input 0

```
5
Nguyen Van A
CNTT2
19/11/2000
3.60
Nguyen Van B
CNTT1
19/3/1999
0.10
Nguyen Van C
DTVT2
21/11/2001
1.30
Nguyen Van D
DTVT2
20/1/2000
2.70
Nguyen Van E
DTVT2
17/5/2003
0.30
```

Sample Output 0

```
SV001 Nguyen Van A CNTT2 19/11/2000 3.60
SV002 Nguyen Van B CNTT1 19/03/1999 0.10
SV003 Nguyen Van C DTVT2 21/11/2001 1.30
SV004 Nguyen Van D DTVT2 20/01/2000 2.70
SV005 Nguyen Van E DTVT2 17/05/2003 0.30
```

[OOP]. Bài 23. Sinh viên 2

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Sinh Viên gồm các thông tin: Mã SV, Họ tên, Ngày sinh, Lớp và Điểm GPA (dạng số thực). Hàm khởi tạo không có tham số, gán các giá trị thuộc tính ở trạng thái mặc định (xâu ký tự rỗng, giá trị số bằng 0). Đọc thông tin N thí sinh từ bàn phím (không có mã sinh viên) và in ra lần lượt màn hình mỗi dòng 1 sinh viên theo đúng thứ tự ban đầu. Trong đó Mã SV được tự tạo ra theo quy tắc thêm mã SV sau đó là giá trị nguyên tự động tăng tính từ 001 (tối đa là 099). Họ tên được xử lý đưa về dạng chuẩn (viết hoa chữ cái đầu của từng từ trong tên). Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy

Input Format

Dòng đầu tiên ghi số sinh viên N ($0 < N < 50$). Mỗi sinh viên ghi trên 4 dòng lần lượt là Họ tên, Lớp, Ngày sinh và Điểm GPA. Trong đó: • Họ tên không quá 30 chữ cái. • Lớp là một chuỗi không có dấu cách. • Ngày sinh có đủ 3 phần ngày tháng năm nhưng có thể chưa đúng chuẩn dd/mm/yyyy. • Điểm GPA đảm bảo trong thang điểm 4 với 2 nhiều nhất 2 số sau dấu phẩy.

Constraints

N/A

Output Format

Ghi ra danh sách lần lượt các sinh viên có đầy đủ Mã sinh viên, Họ tên, Lớp, Ngày sinh (đã chuẩn hóa), điểm GPA (với đúng 2 số sau dấu phẩy). Mỗi sinh viên ghi trên 1 dòng, mỗi thông tin cách nhau 1 khoảng trống.

Sample Input 0

```
5
Hoang Van Long
CNTT2
26/7/2000
0.20
Tran Xuan Loc
CNTT1
3/1/1999
2.40
```

Nguyen Van Manh
CNTT3
18/11/2000
3.30
Ho Sy Kien
CNTT3
4/6/2000
1.50
Le Van Manh
CNTT1
13/3/2002
3.10

Sample Output 0

SV001 Hoang Van Long CNTT2 26/07/2000 0.20
SV002 Tran Xuan Loc CNTT1 03/01/1999 2.40
SV003 Nguyen Van Manh CNTT3 18/11/2000 3.30
SV004 Ho Sy Kien CNTT3 04/06/2000 1.50
SV005 Le Van Manh CNTT1 13/03/2002 3.10

[OOP]. Bài 24. Sắp xếp sinh viên theo GPA và Mã sinh viên

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Viết chương trình khai báo lớp Sinh Viên gồm các thông tin:

- Mã SV
- Họ tên
- Ngày sinh
- Lớp
- Điểm GPA (dạng số thực).

Đọc thông tin N thí sinh từ bàn phím (không nhập mã sinh viên) sau đó sắp xếp theo điểm GPA giảm dần và in ra lần lượt màn hình mỗi dòng 1 sinh viên. Nếu 2 bạn có cùng GPA thì bạn nào có mã sinh viên nhỏ hơn sẽ xếp trước.

Trong đó Mã SV được tự tạo ra theo quy tắc thêm mã SV sau đó là giá trị nguyên tự động tăng tính từ 001 (tối đa là 050). Họ tên được xử lý đưa về dạng chuẩn. Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy

Input Format

Dòng đầu tiên ghi số sinh viên N ($0 < N < 50$). Mỗi sinh viên ghi trên 4 dòng lần lượt là Họ tên, Lớp, Ngày sinh và Điểm GPA. Trong đó: - Họ tên không quá 30 chữ cái.

- Lớp là một chuỗi kí tự không có dấu cách.
- Ngày sinh có đủ 3 phần ngày tháng năm nhưng có thể chưa đúng chuẩn dd/mm/yyyy.
- Điểm GPA đảm bảo trong thang điểm 4 với 2 nhiều nhất 2 số sau dấu phẩy.

Constraints

N/A

Output Format

Ghi ra danh sách lần lượt các sinh viên có đầy đủ Mã sinh viên, Họ tên, Lớp, Ngày sinh (đã chuẩn hóa), điểm GPA (với đúng 2 số sau dấu phẩy) đã được sắp xếp theo điểm GPA giảm dần. Mỗi sinh viên ghi trên 1 dòng, mỗi thông tin cách nhau 1 khoảng trống.

Sample Input 0

```
2
ngUYen Van NaM
CNTT1
2/12/1994
2.17
Nguyen QuanG hai
CNTT2
1/9/1994
3.0
```

Sample Output 0

```
SV002 Nguyen Quang Hai CNTT2 01/09/1994 3.00
SV001 Nguyen Van Nam CNTT1 02/12/1994 2.17
```

[OOP]. Bài 25. Nhân viên 1

- [Problem](#)
- [Submissions](#)

- [Discussions](#)

Một nhân viên làm việc trong công ty được lưu lại các thông tin sau: • Mã nhân viên: được gán tự động tăng, bắt đầu từ 00001 • Họ tên: Xâu ký tự không quá 40 chữ cái. • Giới tính: Nam hoặc Nu • Ngày sinh: đúng theo chuẩn dd/mm/yyyy • Địa chỉ: Xâu ký tự không quá 100 chữ cái • Mã số thuế: Dãy số có đúng 10 chữ số • Ngày ký hợp đồng: đúng theo chuẩn dd/mm/yyyy Viết chương trình nhập danh sách nhân viên (không nhập mã) trong đó có sử dụng chồng toán tử nhập/xuất và in ra màn hình danh sách vừa nhập.

Input Format

Dòng đầu ghi số N là số nhân viên (không quá 40). Mỗi nhân viên ghi trên 6 dòng lần lượt ghi các thông tin theo thứ tự đã ghi trong đề bài. Không có mã nhân viên.

Constraints

N/A

Output Format

Ghi ra danh sách đầy đủ nhân viên, mỗi nhân viên trên một dòng, các thông tin cách nhau đúng một khoảng trống. Thông tin nhân viên được chuẩn hóa ngày sinh và ngày kí hợp đồng.

Sample Input 0

```
3
Nguyen Van A
Nam
22/10/1982
Mo Lao-Ha Dong-Ha Noi
8333012345
31/12/2013
Ly Thi B
Nu
15/10/1988
Mo Lao-Ha Dong-Ha Noi
8333012346
22/08/2011
Hoang Thi C
Nu
04/02/1981
Mo Lao-Ha Dong-Ha Noi
8333012347
22/08/2011
```

Sample Output 0

```
00001 Nguyen Van A Nam 22/10/1982 Mo Lao-Ha Dong-Ha Noi 8333012345 31/12/2013
00002 Ly Thi B Nu 15/10/1988 Mo Lao-Ha Dong-Ha Noi 8333012346 22/08/2011
```

[OOP]. Bài 26. Nhân viên 2

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Một nhân viên làm việc trong công ty được lưu lại các thông tin sau: • Mã nhân viên: được gán tự động tăng, bắt đầu từ 00001. • Họ tên: Xâu ký tự không quá 40 chữ cái. • Giới tính: Nam hoặc Nu. • Ngày sinh: đúng theo chuẩn dd/mm/yyyy. • Địa chỉ: Xâu ký tự không quá 100 chữ cái. • Mã số thuế: Dãy số có đúng 10 chữ số. • Ngày ký hợp đồng: đúng theo chuẩn dd/mm/yyyy. Viết chương trình nhập danh sách nhân viên (không nhập mã) sau đó sắp xếp theo thứ tự ngày sinh từ già nhất đến trẻ nhất và in ra màn hình danh sách đối tượng nhân viên đã sắp xếp. Nếu có 2 nhân viên có cùng ngày sinh thì in ra nhân viên có mã nhân viên nhỏ hơn.

Input Format

Dòng đầu ghi số N là số nhân viên (không quá 40). Mỗi nhân viên ghi trên 6 dòng lần lượt ghi các thông tin theo thứ tự đã ghi trong đề bài. Không có mã nhân viên. Thông tin về ngày sinh và ngày kí hợp đồng được chuẩn hóa.

Constraints

N/A

Output Format

Ghi ra danh sách đầy đủ nhân viên đã sắp xếp, mỗi nhân viên trên một dòng, các thông tin cách nhau đúng một khoảng trống.

Sample Input 0

```
3
Nguyen Van A
Nam
17/10/1995
Bac Ninh
9778128747
6/11/2022
Nguyen Thi B
Nu
18/09/1997
Hai Duong
```


7823490168
5/4/2022
Nguyen Thi C
Nu
02/04/1995
Ninh Binh
7400512775
27/1/2020

Sample Output 0

00003 Nguyen Thi C Nu 02/04/1995 Ninh Binh 7400512775 27/01/2020
00001 Nguyen Van A Nam 17/10/1995 Bac Ninh 9778128747 06/11/2022
00002 Nguyen Thi B Nu 18/09/1997 Hai Duong 7823490168 05/04/2022

[OOP]. Bài 27. Tính giờ quán net

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Quán Game mùa này vắng khách nên chủ quán quyết định tính tiền chi tiết đến từng phút. Dựa trên dữ liệu giờ vào và giờ ra, hãy tính thời gian chơi game của các Game thủ nhé.

Input Format

Dòng đầu của dữ liệu vào ghi số lượng game thủ trong ngày (không quá 20). Thông tin về một game thủ đến chơi game được ghi lại trên 4 dòng lần lượt là:

- username (xâu ký tự độ dài không quá 20, không có khoảng trống).
- password (xâu ký tự độ dài không quá 30, không có khoảng trống).
- Tên người chơi (xâu ký tự độ dài không quá 100, có thể có khoảng trống).
- Giờ vào (định dạng hh:mm)
- Giờ ra (định dạng hh:mm).

Constraints

- $0 \leq \text{hh} \leq 23$
- $0 \leq \text{mm} \leq 59$
- Giờ vào nhỏ hơn giờ ra

Output Format

Ghi ra danh sách game thủ đã được sắp xếp theo thời gian chơi game giảm dần, nếu có 2 game thủ có cùng giờ chơi game thì bạn nào có username có thứ tự từ điển nhỏ hơn được in ra trước.

Sample Input 0

```
5
anhtuanvip
123
Nguyen Van Tuan
05:18
07:06
chickenzero
124
Nguyen Van Phuc
05:38
14:19
anhhung123
matkhau
Nguyen Manh Hung
06:58
14:18
loveyou
acb
Luong Van Manh
02:01
06:47
taikhoan123
matkhaumanh
Nguyen Thi Uyen
00:09
06:19
```

Sample Output 0

```
chickenzero 124 Nguyen Van Phuc 8 gio 41 phut
anhhung123 matkhau Nguyen Manh Hung 7 gio 20 phut
taikhoan123 matkhaumanh Nguyen Thi Uyen 6 gio 10 phut
loveyou acb Luong Van Manh 4 gio 46 phut
anhtuanvip 123 Nguyen Van Tuan 1 gio 48 phut
```

Sample Input 1

```
5
bHaqxvui
iGYZMmhekYzBq
PaGc VdA UaVWtt IsPt DhgKH
09:50
18:30
pGZKupUVGMeI
aADAgxFMmMvUn
kPmi OQt hnMIp NzYHS
```

02:24
12:18
LnIHflzsUEz
FLlpZRmPWpROGF
tpsFu gbBtEt wlvk GGF INg
04:31
07:10
FfqqlBTOB
vKFBIHmBhR
VgFVPZ sju YqOZeV qnjxcO
00:31
03:05
aUuZXfMAx
cWcgqrKmq
muTd HZDtO EOxANe LtuTy SGNA
11:52
13:42

Sample Output 1

pGZKupUVGMeI aADAgxFMmMvUn kPmi OQt hnMIp NzYHS 9 gio 54 phut
bHaqxvui iGYZMmhekYzBq PaGc VdA UaVWtt IsPt DhgKH 8 gio 40 phut
LnIHflzsUEz FLlpZRmPWpROGF tpsFu gbBtEt wlvk GGF INg 2 gio 39 phut
FfqqlBTOB vKFBIHmBhR VgFVPZ sju YqOZeV qnjxcO 2 gio 34 phut
aUuZXfMAx cWcgqrKmq muTd HZDtO EOxANe LtuTy SGNA 1 gio 50 phut

[OOP]. Bài 28. Bảng điểm

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Trường THCS XYZ lập bảng điểm tổng kết cho học sinh. Có 10 môn học lần lượt gồm: Toán, Tiếng Việt, Ngoại ngữ, Vật lý, Hóa học, Sinh học, Lịch Sử, Địa, Giáo dục công dân và môn Công nghệ. Trong đó các môn đều có hệ số là 1. Học sinh được xếp hạng theo điểm trung bình: • Từ 9 trở lên: loại XUẤT SAC. • Từ 8 đến 8.9: loại GIOI. • Từ 7 đến 7.9: loại KHA. • Từ 5 đến 6.9: loại TB. • Dưới 5: loại YEU Hãy lập bảng điểm tổng kết và sắp xếp theo điểm trung bình giảm dần.

Input Format

Dòng đầu ghi số học sinh (không quá 50). Thông tin về mỗi học sinh có hai dòng: dòng đầu là họ tên (độ dài không quá 50), dòng thứ 2 gồm 10 số thực trong đoạn [0..10] lần lượt là điểm 10 môn theo đúng thứ tự đã mô tả.

Constraints

Chú ý : Sau điểm cuối cùng có dấu cách :)

Output Format

Danh sách đã sắp xếp được ghi ra bao gồm các thông tin: • Mã học sinh (tự động gán tăng dần theo thứ tự nhập, bắt đầu là HS01, chú ý bài này học sinh có mã 10 sẽ có mã học sinh là SV010) • Họ và tên • Điểm trung bình (với 1 chữ số phần thập phân) • Xếp loại Trong trường hợp điểm trung bình bằng nhau thì học sinh nào có mã học sinh nhỏ hơn sẽ xếp trên.

Sample Input 0

```
5
Tran Duc Luong
7.8 5.9 9.0 7.2 6.5 4.6 8.3 9.6 4.0 6.0
Do Xuan Loc
8.4 0.9 4.4 4.6 2.4 9.5 2.1 3.6 9.5 6.3
Vu Duc Hieu
6.5 8.8 7.7 9.4 2.9 0.5 1.5 3.7 4.8 6.9
Phan Van Nam
8.2 6.4 8.3 2.3 8.3 2.9 8.7 7.1 9.4 3.4
Vu Duc Minh
7.8 5.3 8.3 1.6 2.6 9.8 0.6 0.1 5.2 2.6
```

Sample Output 0

```
HS01 Tran Duc Luong 6.9 TB
HS04 Phan Van Nam 6.5 TB
HS03 Vu Duc Hieu 5.3 TB
HS02 Do Xuan Loc 5.2 TB
HS05 Vu Duc Minh 4.4 YEU
```

[OOP]. Bài 29. Sắp xếp sinh viên theo lớp

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Thông tin về mỗi sinh viên gồm: • Mã sinh viên: dãy ký tự không có khoảng trống (không quá 15). Đảm bảo không trùng nhau. • Họ và tên: độ dài không quá 100. • Lớp: dãy ký tự không có khoảng trống (không quá 15). • Email: dãy ký tự không có khoảng trống (không quá 15). Hãy nhập danh sách sinh viên và sắp xếp theo lớp tăng dần (thứ tự từ điển).

Input Format

Dòng đầu ghi số sinh viên. Mỗi sinh viên ghi trên 4 dòng lần lượt là: mã, họ tên, lớp, email. Có không quá 5000 sinh viên trong danh sách.

Constraints

N/A

Output Format

Ghi ra danh sách sinh viên đã sắp xếp theo lớp. Mỗi sinh viên trên một dòng, các thông tin cách nhau một khoảng trống. Nếu 2 sinh viên có cùng lớp thì sắp xếp theo mã tăng dần (thứ tự từ điển)

Sample Input 0

```
4
SV001
Nguyen Trong Duc Anh
CNTT1
sv1@gmail.com
SV002
To Ngoc Hieu
CNTT2
sv2@gmail.com
SV003
Nguyen Ngoc Son
HTTT2
sv3@gmail.com
SV004
Nguyen Trong Tung
CNTT1
sv4@gmail.com
```

Sample Output 0

```
SV001 Nguyen Trong Duc Anh CNTT1 sv1@gmail.com
SV004 Nguyen Trong Tung CNTT1 sv4@gmail.com
SV002 To Ngoc Hieu CNTT2 sv2@gmail.com
SV003 Nguyen Ngoc Son HTTT2 sv3@gmail.com
```

[OOP]. Bài 30. Sắp xếp sinh viên theo mã

- [Problem](#)
- [Submissions](#)

- [Discussions](#)

Thông tin về mỗi sinh viên gồm: • Mã sinh viên: dãy ký tự không có khoảng trống (không quá 15). Đảm bảo không trùng nhau. • Họ và tên: độ dài không quá 100. • Lớp: dãy ký tự không có khoảng trống (không quá 15). • Email: dãy ký tự không có khoảng trống (không quá 15). Hãy nhập danh sách sinh viên và sắp xếp theo mã sinh viên tăng dần (thứ tự từ điển).

Input Format

Mỗi sinh viên ghi trên 4 dòng lần lượt là: mã, họ tên, lớp, email. Không cho biết số sinh viên nhưng dữ liệu đảm bảo là chẵn lần 4 dòng. Có không quá 5000 sinh viên trong danh sách.

Constraints

N/A

Output Format

Ghi ra danh sách sinh viên đã sắp xếp theo mã. Mỗi sinh viên trên một dòng, các thông tin cách nhau một khoảng trống.

Sample Input 0

```
SV004
Nguyen Trong Tung
CNTT1
sv4@gmail.com
SV001
Nguyen Trong Duc Anh
CNTT1
sv1@gmail.com
SV002
To Ngoc Hieu
CNTT2
sv2@gmail.com
SV003
Nguyen Ngoc Son
HTTT2
sv3@gmail.com
```

Sample Output 0

```
SV001 Nguyen Trong Duc Anh CNTT1 sv1@gmail.com
SV002 To Ngoc Hieu CNTT2 sv2@gmail.com
SV003 Nguyen Ngoc Son HTTT2 sv3@gmail.com
SV004 Nguyen Trong Tung CNTT1 sv4@gmail.com
```

[OOP]. Bài 31. Liệt kê sinh viên theo lớp

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Thông tin về mỗi sinh viên gồm: • Mã sinh viên: dãy ký tự không có khoảng trống (không quá 15). Đảm bảo không trùng nhau. • Họ và tên: độ dài không quá 100, chưa được chuẩn hóa. • Lớp: dãy ký tự không có khoảng trống (không quá 15). • Email: dãy ký tự không có khoảng trống (không quá 15). Hãy nhập danh sách sinh viên và liệt kê sinh viên theo lớp.

Input Format

Dòng đầu ghi số sinh viên (không quá 1000). Mỗi sinh viên ghi trên 4 dòng lần lượt là: mã, họ tên, lớp, email. Sau đó sẽ có giá trị số Q là số truy vấn. Tiếp theo là Q dòng, mỗi dòng ghi một lớp.

Constraints

$1 \leq Q \leq 20$;

Output Format

Với mỗi truy vấn, liệt kê danh sách sinh viên của lớp đó theo mẫu như trong ví dụ. Mỗi sinh viên ghi trên một dòng, các thông tin cách nhau một khoảng trống. Thứ tự sinh viên vẫn giữ nguyên như thứ tự ban đầu.

Sample Input 0

```
4
20132238
hoang dinh NAm
CNTT1
namhd@gmail.com
20142231
PhAM nGoc tHuan
HTTT2
thuanpn@gmail.com
20148971
Nguyen duc XUAN
CNTT1
xuannd@gmail.com
20132038
```

PhAM vAn Hung
hungpv@gmail.com
CNTT2
1
CNTT1

Sample Output 0

DANH SÁCH SINH VIÊN LỚP CNTT1 :
20132238 Hoang Dinh Nam CNTT1 namhd@gmail.com
20148971 Nguyen Duc Xuan CNTT1 xuannd@gmail.com

[OOP]. Bài 32. Liệt kê sinh viên theo khóa

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Thông tin về mỗi sinh viên gồm: - Mã sinh viên: dãy ký tự không có khoảng trống (không quá 15). Đảm bảo không trùng nhau.

- Họ và tên: độ dài không quá 100, chưa được chuẩn hóa và cần được chuẩn hóa trước khi in.
- Lớp: dãy ký tự không có khoảng trống (không quá 15).
- Email: dãy ký tự không có khoảng trống (không quá 15).

Hãy nhập danh sách sinh viên và liệt kê sinh viên theo khóa học. Chú ý: dữ liệu khóa học thể hiện qua 4 chữ số đầu tiên trong mã sinh viên

Input Format

- Dòng đầu ghi số sinh viên (không quá 1000).
- Mỗi sinh viên ghi trên 4 dòng lần lượt là: mã, họ tên, lớp, email.
- Sau đó sẽ có giá trị số Q là số truy vấn.
- Tiếp theo là Q dòng, mỗi dòng ghi năm bắt đầu khóa học theo định dạng yyyy.

Constraints

$1 \leq Q \leq 100$

Output Format

Với mỗi truy vấn, liệt kê danh sách sinh viên của khóa đó theo mẫu như trong ví dụ. Mỗi sinh viên ghi trên một dòng, các thông tin cách nhau một khoảng trống. Thứ tự sinh viên vẫn giữ nguyên như thứ tự ban đầu.

Sample Input 0

```
4
20132238
hoang dinh NAM
CNTT1
namhd@gmail.com
20142231
PhAM nGoc tHuan
HTTT2
thuanpn@gmail.com
20148971
Nguyen duc XUAN
CNTT1
xuannd@gmail.com
20132038
PhAM vAn Hung
CNTT2
hungpv@gmail.com
2
2013
2014
```

Sample Output 0

```
DANH SACH SINH VIEN KHOA 2013 :
20132238 Hoang Dinh Nam CNTT1 namhd@gmail.com
20132038 Pham Van Hung CNTT2 hungpv@gmail.com
DANH SACH SINH VIEN KHOA 2014 :
20142231 Pham Ngoc Thuan HTTT2 thuanpn@gmail.com
20148971 Nguyen Duc Xuan CNTT1 xuannd@gmail.com
```

[OOP]. Bài 33. Liệt kê sinh viên theo ngành

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Thông tin về mỗi sinh viên gồm: • Mã sinh viên: dãy ký tự không có khoảng trống (không quá 15). Đảm bảo không trùng nhau. • Họ và tên: độ dài không quá 100.

• Lớp: dãy ký tự không có khoảng trống (không quá 15). • Email: dãy ký tự không có khoảng trống (không quá 15). Hãy nhập danh sách sinh viên và liệt kê sinh viên theo ngành. Ngành học của sinh viên sẽ dựa trên 4 chữ cái đầu tiên của lớp bao gồm các ngành : CNTT : Công nghệ thông tin. DTVT : Điện tử viễn thông. KT : Kế toán. MKT : Marketing.

Input Format

Dòng đầu ghi số sinh viên (không quá 1000) Mỗi sinh viên ghi trên 4 dòng lần lượt là: mã, họ tên, lớp, email. Sau đó sẽ có giá trị số Q là số truy vấn. Mỗi truy vấn sẽ là 1 trong 4 ngành nêu trên.

Constraints

$$1 \leq q \leq 100$$

Output Format

Với mỗi truy vấn, liệt kê danh sách sinh viên của ngành đó theo mẫu như trong ví dụ. Mỗi sinh viên ghi trên một dòng, các thông tin cách nhau một khoảng trống. Thứ tự sinh viên vẫn giữ nguyên như thứ tự ban đầu

Sample Input 0

```
3
20211000
Nguyen Ngoc Huong
CNTT2
DqFvSInO@gmail.com
20181001
Vu Duc Huong
CNTT2
KBvHj@gmail.com
20191002
Pham Duc Hoang
DTVT2
HEKglq@gmail.com
1
CNTT
```

Sample Output 0

```
DANH SACH SINH VIEN NGANH CONG NGHE THONG TIN :
20211000 Nguyen Ngoc Huong CNTT2 DqFvSInO@gmail.com
20181001 Vu Duc Huong CNTT2 KBvHj@gmail.com
```

[OOP]. Bài 34. Sắp xếp giảng viên

- Problem

- [Submissions](#)
- [Discussions](#)

Danh sách giảng viên của trường đại học cần được sắp xếp lại theo tên. Thông tin về giảng viên ban đầu chỉ có họ tên và Bộ môn. Mã giảng viên tự động tăng, tính từ GV01. Cần sắp xếp lại theo tên (tức là từ cuối cùng trong xâu họ tên). Các giảng viên có cùng tên thì được sắp xếp theo mã giảng viên.

Input Format

Dòng đầu ghi số giảng viên (không quá 50). Mỗi giảng viên ghi trên 2 dòng gồm họ tên (không quá 50 ký tự) và Bộ môn (không quá 30 ký tự).

Constraints

N/A

Output Format

Danh sách đã sắp xếp trong đó mỗi giảng viên ghi trên một dòng. Mã được tự động điền theo thứ tự nhập, bộ môn được viết tắt theo các chữ cái đầu của từng từ và ở dạng in hoa.

Sample Input 0

```
5
Tran Duc Huong
Toan tin
Vu Ngoc Manh
Da phuong tien
Tran Van Huong
Cong nghe thong tin
Phan Xuan Minh
Ke toan
Phan Ngoc Huong
Ke toan
```

Sample Output 0

```
GV01 Tran Duc Huong TT
GV03 Tran Van Huong CNTT
GV05 Phan Ngoc Huong KT
GV02 Vu Ngoc Manh DPT
GV04 Phan Xuan Minh KT
```