

## STRUCT

Bài 1. Phân số.....	2
Bài 2. Tính tổng 2 phân số .....	2
Bài 3. Cấu trúc điểm.....	3
Bài 4. Cấu trúc điểm.....	4
Bài 5. Cấu trúc sinh viên .....	5
Bài 6. Cấu trúc nhân viên .....	6
Bài 7. Danh sách nhân viên.....	7
Bài 8. Diện tích hình tròn ngoại tiếp tam giác .....	9
Bài 9. Số thuận nghịch giảm dần .....	11
Bài 10. Trẻ nhất, già nhất .....	12
Bài 11. Bảng điểm thành phần 1 .....	12
Bài 12. Bảng điểm thành phần 2 .....	14
Bài 13. Sắp xếp thời gian .....	15
Bài 14. Danh sách thực tập 1.....	16
Bài 15. Danh sách thực tập 2.....	18
Bài 16. Phép toán với phân số.....	21
Bài 17. Danh sách sinh viên 1 .....	22
Bài 18. Danh sách sinh viên 2.....	23
Bài 19. Danh sách sinh viên 3 .....	25
Bài 20. Sắp xếp danh sách mặt hàng.....	26

## Bài 1. Phân số

Viết chương trình xây dựng cấu trúc Phân số gồm hai phần tử là tử số và mẫu số. Các giá trị đều nguyên dương và không quá 18 chữ số.

Sau đó thực hiện nhập vào một phân số và in ra phân số đó ở dạng tối giản.

### Input

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

### Output

Ghi ra phân số tối giản như trong ví dụ

### Ví dụ

Input	Output
123 456	41/152

Sử dụng hàm main có sẵn như sau

```
int main() {  
    struct PhanSo p;  
    nhap(p);  
    rutgon(p);  
    in(p);  
    return 0;  
}
```

Source : <https://ideone.com/FzqPmc>

## Bài 2. Tính tổng 2 phân số

Viết chương trình xây dựng cấu trúc Phân số gồm hai phần tử là tử số và mẫu số. Các giá trị đều nguyên dương và không quá 9 chữ số.

Sau đó thực hiện nhập vào hai phân số p và q. Tính tổng  $p + q$ , rút gọn và in ra kết quả.

### Input

Có bốn số nguyên dương lần lượt là tử số và mẫu số của p rồi đến q.

## Output

Ghi ra phân số tổng  $p + q$  ở dạng tối giản như trong ví dụ

## Ví dụ

Input	Output
123 456 12 34	1609/2584

Hàm main có sẵn như sau:

```
int main() {  
    struct PhanSo p,q;  
    nhap(p); nhap(q);  
    PhanSo t = tong(p,q);  
    in(t);  
    return 0;  
}
```

Source : <https://ideone.com/a89NeK>

## Bài 3. Cấu trúc điểm

Viết chương trình khai báo cấu trúc điểm trong không gian hai chiều với hai tọa độ  $x$  và  $y$  (kiểu số thực double).

```
struct Point{  
    double x, y;  
}
```

Nhập hai điểm A và B. Tính khoảng cách giữa hai điểm A, B với độ chính xác 4 chữ số sau dấu phẩy.

## Input

- Dòng đầu ghi số bộ test, không quá 20.
- Mỗi bộ test có 4 số thực lần lượt là tọa độ của 2 điểm A và B, giá trị tuyệt đối không quá 1000.

## Output

Với mỗi bộ test, viết ra khoảng cách giữa 2 điểm với 4 chữ số sau dấu phẩy.

### Ví dụ

Input	Output
2	5.0000
0 0 0 5	193.0648
0 199 5 6	

Yêu cầu hàm main có sẵn :

```
int main(){
    struct Point A, B;
    int t;
    cin>>t;
    while(t--){
        input(A); input(B);
        cout << fixed << setprecision(4) << distance(A,B) << endl;
    }
    return 0;
}
```

Source : <https://ideone.com/Zjm8gR>

### Bài 4. Cấu trúc điểm

Viết chương trình khai báo cấu trúc Thí Sinh gồm các thông tin: Họ tên, 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

Gồm 5 dòng lần lượt, mỗi dòng ghi 1 thông tin: Họ tên, Ngày sinh, Điểm môn 1, Điểm môn 2, Điểm môn 3. Họ tên không quá 50 chữ cái, Ngày sinh viết đúng chuẩn dd/mm/yyyy. Các giá trị điểm là số thực (float).

### Output

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.

### Ví dụ

Input	Output
Nguyen Hoang Ha 11/10/2001 4.5 10.0 5.5	Nguyen Hoang Ha 11/10/2001 20.0

Hàm main có sẵn :

```
int main(){  
    struct ThiSinh A;  
    nhap(A);  
    in(A);  
    return 0;  
}
```

Source : <https://ideone.com/SWomcA>

### Bài 5. Cấu trúc sinh viên

Viết chương trình khai báo cấu trúc Sinh Viên gồm các thông tin: Mã SV, Họ tên, Lớp, Ngày sinh và Điểm GPA (dạng số thực float).

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 **B20DCCN001**. Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy.

### Input

Gồm 4 dòng lần lượt là Họ tên, Lớp, Ngày sinh và Điểm GPA.

Trong đó:

- Họ tên không quá 30 chữ cái.
- Lớp theo đúng định dạng thường dùng ở PTIT

- 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.

## Output

Ghi thông tin sinh viên trên 1 dòng, mỗi thông tin cách nhau 1 khoảng trống.

## Ví dụ

Input	Output
Nguyen Hoa Binh D20CQCN04-B 2/2/2002 2	B20DCCN001 Nguyen Hoa Binh D20CQCN04-B 02/02/2002 2.00

Hàm main có sẵn :

```
int main(){
    struct SinhVien a;
    nhap(a);
    in(a);
    return 0;
}
```

Source : <https://ideone.com/X8SZwM>

## Bài 6. Cấu trúc nhân viên

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

- Mã nhân viên: được gán giá trị là 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 một nhân viên (không nhập mã) và in ra màn hình thông tin của nhân viên đó.

### Input

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

### Output

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

### Ví dụ

Input
Nguyen Van Hoa
Nam
11/22/1982
Mo Lao-Ha Dong-Ha Noi
8333123456
31/12/2013
Output
00001 Nguyen Van Hoa Nam 11/22/1982 Mo Lao-Ha Dong-Ha Noi 8333123456 31/12/2013

Hàm main có sẵn:

```
int main(){  
    struct NhanVien a;  
    nhap(a);  
    in(a);  
    return 0;  
}
```

Source : <https://ideone.com/fODmKA>

### Bài 7. Danh sách nhân viên

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) và in ra màn hình danh sách nhân viên vừa nhập.

### Input

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.

### Output

Ghi ra danh sách đầy đủ nhân viên theo đúng thứ tự nhập, mỗi nhân viên trên một dòng, các thông tin cách nhau đúng một khoảng trống.

Ví dụ

Input
3
Nguyen Van A
Nam
10/22/1982
Mo Lao-Ha Dong-Ha Noi
8333012345
31/12/2013
Ly Thi B
Nu
10/15/1988



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

---

### Output

00001 Nguyen Van A Nam 10/22/1982 Mo Lao-Ha Dong-Ha Noi 8333012345 31/12/2013

00002 Ly Thi B Nu 10/15/1988 Mo Lao-Ha Dong-Ha Noi 8333012346 22/08/2011

00003 Hoang Thi C Nu 04/02/1981 Mo Lao-Ha Dong-Ha Noi 8333012347 22/08/2011

---

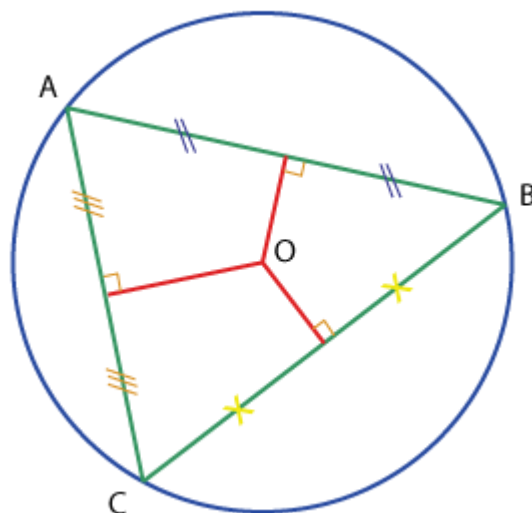
Hàm main có sẵn :

```
int main(){  
  
    struct NhanVien ds[50];  
  
    int N,i;  
  
    cin >> N;  
  
    for(i = 0; i < N; i++) nhap(ds[i]);  
  
    inds(ds,N);  
  
    return 0;  
  
}
```

Source : <https://ideone.com/hr7gpG>

Bài 8. Diện tích hình tròn ngoại tiếp tam giác  
Cho ba điểm A, B, C trong không gian hai chiều 0xy.

Hãy tính diện tích hình tròn ngoại tiếp tam giác tạo bởi 3 điểm trên.



Công thức Heron tính diện tích tam giác với 3 cạnh là a, b, c:

$$S = \frac{1}{4} \sqrt{(a+b+c)(a+b-c)(b+c-a)(c+a-b)}$$

Công thức tính bán kính hình tròn ngoại tiếp:

$$R = \frac{a.b.c}{4S}$$

Khi tính diện tích nên dùng hằng số PI

```
#define PI 3.141592653589793238
```

### Input

Dòng đầu ghi số bộ test (không quá 20).

Mỗi bộ test ghi trên 1 dòng 6 số thực lần lượt là tọa độ của 3 điểm A, B, C. Giá trị tọa độ không quá 1000.

### Output

Nếu 3 điểm không thể tạo thành tam giác, in ra INVALID

Nếu 3 điểm tạo thành tam giác, in ra diện tích hình tròn ngoại tiếp với độ chính xác 3 số phần thập phân.

#### Ví dụ

Input	Output
3	INVALID
0 0 0 5 0 199	INVALID
1 1 1 1 1 1	39.270
0 0 0 5 5 0	

Source : <https://ideone.com/3v1iv0>

#### Bài 9. Số thuận nghịch giảm dần

Cho dữ liệu vào dạng văn bản, với không quá 1000 số nguyên dương, các số không quá 50 chữ số. Hãy liệt kê các số thuận nghịch khác nhau theo thứ tự giảm dần và số lần xuất hiện của nó.

#### Input

Luồng vào dạng văn bản với không quá 1000 số nguyên dương. Không biết trước số dòng hoặc số lượng chính xác bao nhiêu số nguyên.

#### Output

Ghi ra các số thuận nghịch theo thứ tự giá trị giảm dần và số lần xuất hiện của nó, mỗi số trên một dòng. Không tính các số có 1 chữ số.

#### Ví dụ

Input	Output
12321 456 12321 34 56 999999999999999999	999999999999999999 1
12 3 43 34 54 34 54 34 54 657	12321 4
5 45 554 12321 12321	565 3
65 76 45 45 34 53	
546 565 65645 6 65 65 35 65	
565 565	

Source : <https://ideone.com/THn8pu>

<https://ideone.com/UkuTNW>

### Bài 10. Trẻ nhất, già nhất

Cho một danh sách tên người và ngày tháng năm sinh.

Hãy tìm ra người trẻ nhất và người già nhất.

#### Input

Dòng 1 ghi số N là số người (không quá 100).

N dòng tiếp theo, mỗi dòng ghi tên (xâu ký tự không có khoảng trống và không quá 15 ký tự, sau đó là dãy ký tự mô tả ngày tháng năm sinh theo chuẩn dd/mm/yyyy).

#### Output

Dòng đầu ghi ra tên người trẻ nhất.

Dòng thứ 2 ghi ra tên người già nhất.

Dữ liệu đảm bảo không có 2 người nào trùng ngày sinh.

#### Ví dụ

Input	Output
5	Binh
Nam 01/10/1991	Tam
An 30/12/1990	
Binh 15/08/1993	
Tam 18/09/1990	
Truong 20/09/1990	

Source : <https://ideone.com/ujI3AC>

### Bài 11. Bảng điểm thành phần 1

Cho dữ liệu bảng điểm thành phần trong đó thông tin của mỗi sinh viên gồm:

- Mã sinh viên (xâu ký tự độ dài không quá 15, không có khoảng trống)
- Tên sinh viên (xâu ký tự, độ dài không quá 50)
- Lớp (xâu ký tự độ dài không quá 15, không có khoảng trống)
- Điểm 1, Điểm 2, Điểm 3: mỗi điểm là một số thực (hệ 10)

Hãy sắp xếp lại bảng điểm thành phần theo mã sinh viên (thứ tự từ điển tăng dần).

### Input

Dòng đầu ghi số sinh viên (không quá 100).

Mỗi sinh viên ghi trên 6 dòng lần lượt là: Mã SV, Họ tên, Lớp, Điểm 1, Điểm 2, Điểm 3.

### Output

Ghi ra danh sách sinh viên đã sắp xếp theo mã sinh viên.

Mỗi sinh viên ghi trên 1 dòng gồm các thông tin: thứ tự, mã sv, họ tên, lớp, điểm 1, điểm 2, điểm 3. Các thông tin cách nhau đúng một khoảng trống.

Các giá trị điểm ghi ra với đúng 1 chữ số phần thập phân.

### Ví dụ

Input	Output
3	1 B20DCAT001 Le Van Nam D20CQAT02-B 6.0 6.0 4.0
B20DCCN999	2 B20DCCN111 Tran Hoa Binh D20CQCN04-B 9.0 5.0 6.0
Nguyen Van An	3 B20DCCN999 Nguyen Van An D20CQCN0
D20CQCN04-B	
10.0	
9.0	
8.0	
B20DCAT001	
Le Van Nam	
D20CQAT02-B	

6.0	
6.0	
4.0	
B20DCCN111	
Tran Hoa Binh	
D20CQCN04-B	
9.0	
5.0	
6.0	

Source : <https://ideone.com/qlWT2c>

## Bài 12. Bảng điểm thành phần 2

Cho dữ liệu bảng điểm thành phần trong đó thông tin của mỗi sinh viên gồm:

- Mã sinh viên (xâu ký tự độ dài không quá 15, không có khoảng trống)
- Họ tên sinh viên (xâu ký tự, độ dài không quá 50)
- Lớp (xâu ký tự độ dài không quá 15, không có khoảng trống)
- Điểm 1, Điểm 2, Điểm 3: mỗi điểm là một số thực (hệ 10)

Hãy sắp xếp lại bảng điểm thành phần theo họ tên (thứ tự từ điển tăng dần - so sánh cả xâu ký tự họ tên để sắp xếp, không cần tách tên).

### Input

Dòng đầu ghi số sinh viên (không quá 100).

Mỗi sinh viên ghi trên 6 dòng lần lượt là: Mã SV, Họ tên, Lớp, Điểm 1, Điểm 2, Điểm 3.

### Output

Ghi ra danh sách sinh viên đã sắp xếp theo họ tên.

Mỗi sinh viên ghi trên 1 dòng gồm các thông tin: thứ tự, mã sv, họ tên, lớp, điểm 1, điểm 2, điểm 3. Các thông tin cách nhau đúng một khoảng trống. Các giá trị điểm ghi ra với đúng 1 chữ số phần thập phân.

### Ví dụ

Input	Output
3	1 B20DCAT001 Le Van An D20CQAT02-B 6.0 6.0 4.0
B20DCCN999	2 B20DCCN111 Nguyen Van Binh D20CQCN01-B 9.0 5.0 6.0
Nguyen Van Nam	3 B20DCCN999 Nguyen Van Nam D20CQ
D20CQCN04-B	
10.0	
9.0	
8.0	
B20DCAT001	
Le Van An	
D20CQAT02-B	
6.0	
6.0	
4.0	
B20DCCN111	
Nguyen Van Binh	
D20CQCN01-B	
9.0	
5.0	
6.0	

Source : <https://ideone.com/9yZ4DE>

### Bài 13. Sắp xếp thời gian

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

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).

## Output

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

## Ví dụ

Input	Output
3	11 15 12
11 20 20	11 20 20
14 20 14	14 20 14
11 15 12	

Source : <https://ideone.com/8LBhbg>

## Bài 14. Danh sách thực tập 1

Sinh viên CNTT PTIT đến năm cuối được cử đi thực tập tại các doanh nghiệp.

Thông tin của mỗi sinh viên trong danh sách thực tập bao gồm:

- Số thứ tự: là 1 số nguyên tự động tăng
- Mã sinh viên: là một chuỗi ký tự không có khoảng trống, không quá 12 ký tự
- Họ tên: là một chuỗi ký tự họ tên đã chuẩn hóa, không quá 50 ký tự
- Lớp: là một chuỗi ký tự không có khoảng trống, không quá 10 ký tự
- Email: là một địa chỉ email, không có khoảng trống, không quá 100 ký tự
- Doanh nghiệp: tên viết tắt của doanh nghiệp, không có khoảng trống, không quá 15 ký tự.

Hãy viết chương trình đọc vào danh sách thực tập sau đó in danh sách cho từng doanh nghiệp theo yêu cầu.

## Input

Dòng đầu ghi số N là sinh viên



Mỗi sinh viên ghi trên 5 dòng gồm mã, họ tên, lớp, email và doanh nghiệp.

Không có số thứ tự, cần tự gán theo thứ tự tăng dần từ 1.

Sau khi hết danh sách sinh viên sẽ có một số nguyên Q (không quá 5) cho biết danh sách truy vấn.

Tiếp theo là Q dòng, mỗi dòng ghi tên một doanh nghiệp (đúng như trong danh sách, không có trường hợp nào không tồn tại trong danh sách)

### Output

Với mỗi doanh nghiệp, liệt kê danh sách sinh viên thực tập ở doanh nghiệp đó theo thứ tự sắp xếp họ tên (so sánh cả xâu họ tên theo thứ tự từ điển, không cần tách riêng phần tên).

Mỗi sinh viên trên một dòng. Mỗi thông tin trong danh sách cách nhau đúng một khoảng trống.

### Ví dụ

Input	Output
6	3 B17DCAT092 Cao Danh Huy D17CQAT04-B test3@stu.ptit.edu.vn FPT
B17DCCN016	2 B17DCCN107 Dao Thanh Dat D17CNPM5 test2@stu.ptit.edu.vn FPT
Le Khac Tuan Anh	5 B17DCCN461 Dinh Quang
D17HTTT2	
test1@stu.ptit.edu.vn	
VIETTEL	
B17DCCN107	
Dao Thanh Dat	
D17CNPM5	
test2@stu.ptit.edu.vn	
FPT	
B17DCAT092	

Cao Danh Huy	
D17CQAT04-B	
test3@stu.ptit.edu.vn	
FPT	
B17DCCN388	
Cao Sy Hai Long	
D17CNPM2	
test4@stu.ptit.edu.vn	
VNPT	
B17DCCN461	
Dinh Quang Nghia	
D17CNPM2	
test5@stu.ptit.edu.vn	
FPT	
B17DCCN554	
Bui Xuan Thai	
D17CNPM1	
test6@stu.ptit.edu.vn	
GAMELOFT	
1	
FPT	

Source : <https://ideone.com/nql0bM>

## Bài 15. Danh sách thực tập 2

Sinh viên CNTT PTIT đến năm cuối được cử đi thực tập tại các doanh nghiệp.

Thông tin của mỗi sinh viên trong danh sách thực tập bao gồm:

- Số thứ tự: là 1 số nguyên tự động tăng
- Mã sinh viên: là một xâu ký tự không có khoảng trống, không quá 12 ký tự
- Họ tên: là một xâu ký tự họ tên đã chuẩn hóa, không quá 50 ký tự
- Lớp: là một xâu ký tự không có khoảng trống, không quá 10 ký tự
- Email: là một địa chỉ email, không có khoảng trống, không quá 100 ký tự
- Doanh nghiệp: tên viết tắt của doanh nghiệp, không có khoảng trống, không quá 15 ký tự.

Hãy viết chương trình đọc vào danh sách thực tập sau đó in danh sách cho từng doanh nghiệp theo yêu cầu.

### Input

Dòng đầu ghi số N là sinh viên

Mỗi sinh viên ghi trên 5 dòng gồm mã, họ tên, lớp, email và doanh nghiệp.

Không có số thứ tự, cần tự gán theo thứ tự tăng dần từ 1.

Sau khi hết danh sách sinh viên sẽ có một số nguyên Q (không quá 5) cho biết danh sách truy vấn.

Tiếp theo là Q dòng, mỗi dòng ghi tên một doanh nghiệp (đúng như trong danh sách, không có trường hợp nào không tồn tại trong danh sách)

### Output

Với mỗi doanh nghiệp, liệt kê danh sách sinh viên thực tập ở doanh nghiệp đó theo thứ tự sắp xếp mã sinh viên (so sánh theo thứ tự từ điển).

Mỗi sinh viên trên một dòng. Mỗi thông tin trong danh sách cách nhau đúng một khoảng trống.

### Ví dụ

Input	Output
6	3 B17DCAT092 Cao Danh Huy D17CQAT04-B test3@stu.ptit.edu.vn FPT
B17DCCN016	2 B17DCCN107 Dao Thanh Dat D17CNPM5 test2@stu.ptit.edu.vn FPT
Le Khac Tuan Anh	5 B17DCCN461 Dinh Quang Nghia D17C
D17HTTT2	

test1@stu.ptit.edu.vn

VIETTEL

B17DCCN107

Dao Thanh Dat

D17CNPM5

test2@stu.ptit.edu.vn

FPT

B17DCAT092

Cao Danh Huy

D17CQAT04-B

test3@stu.ptit.edu.vn

FPT

B17DCCN388

Cao Sy Hai Long

D17CNPM2

test4@stu.ptit.edu.vn

VNPT

B17DCCN461

Dinh Quang Nghia

D17CNPM2

test5@stu.ptit.edu.vn

FPT

B17DCCN554

Bui Xuan Thai

D17CNPM1

test6@stu.ptit.edu.vn

GAMELOFT

1

FPT

Source : <https://ideone.com/C8r7sF>

### Bài 16. Phép toán với phân số

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:

- Tính  $C = (A + B)^2$  và rút gọn kết quả.
- Tính  $D = A \times B \times C$  và rút gọn kết quả.

#### Input:

Dòng đầu tiên là số bộ test T ( $T \leq 100$ )

T dòng tiếp theo, mỗi dòng gồm 4 số lần lượt là tử và mẫu số của phân số A và phân số B với  $-10^2 \leq$  tử số, mẫu số  $\leq 10^2$ . Mẫu số là số khác 0.

#### Output:

Kết quả của hai phép tính theo định dạng phân số.

#### Ví dụ

Input:	Output:
2	25/16 75/128
1 2 3 4	484/225 3872/3375
2 3 4 5	

Hàm

main có sẵn

```
int main() {
```

```
    int t;
```

```

cin >> t;

while (t--) {
    PhanSo A;

    PhanSo B;

    cin >> A.tu >> A.mau >> B.tu >> B.mau;

    process(A, B);
}
}

```

Source : <https://ideone.com/eIybVo>

### Bài 17. Danh sách sinh viên 1

Viết chương trình khai báo cấu trúc Sinh Viên gồm các thông tin: Mã SV, Họ tên, Lớp, Ngày sinh và Điểm GPA (dạng số thực float).

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 **B20DCCN** 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

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 theo đúng định dạng thường dùng ở PTIT
- 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.

#### Output

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.

#### Ví dụ

Input	Output
1  Nguyen Van An  D20CQCN01-B  2/12/2002  3.19	B20DCCN001 Nguyen Van An D20CQCN01-B 02/12/2002 3.19

Hàm main có sẵn

```
int main(){
    struct SinhVien ds[50];
    int N;
    cin >> N;
    nhap(ds, N);
    in(ds, N);
    return 0;
}
```

Source : <https://ideone.com/alwXVo>

### Bài 18. Danh sách sinh viên 2

Viết chương trình khai báo cấu trúc Sinh Viên gồm các thông tin: Mã SV, Họ tên, Lớp, Ngày sinh và Điểm GPA (dạng số thực float).

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 **B20DCCN** 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. Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy

### Input

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 theo đúng định dạng thường dùng ở PTIT
- 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.

## Output

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.

## Ví dụ

Input	Output
1  nGuyEn vaN biNH  D20CQCN01-B  2/12/2002  3.1	B20DCCN001 Nguyen Van Binh D20CQCN01-B 02/12/2002 3.10

Hàm main có sẵn

```
int main(){
    struct SinhVien ds[50];
    int N;
    cin >> N;
    nhap(ds, N);
    in(ds, N);
    return 0;
}
```

Source : <https://ideone.com/50Cj1h>



### Bài 19. Danh sách sinh viên 3

Viết chương trình khai báo cấu trúc Sinh Viên gồm các thông tin: Mã SV, Họ tên, Ngày sinh, Lớp và Điểm GPA (dạng số thực float).

Đọc thông tin N sinh viên từ bàn phím (không có mã sinh viên) sau đó sắp xếp theo điểm GPA giảm dần và in ra lần lượt màn hình mỗi dòng 1 sinh viên.

Trong đó Mã SV được tự tạo ra theo quy tắc thêm mã **B20DCCN** 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. Ngày sinh được chuẩn hóa về dạng dd/mm/yyyy

#### Input

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 theo đúng định dạng thường dùng ở PTIT
- 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.

Dữ liệu đảm bảo không có hai sinh viên nào có điểm GPA bằng nhau.

#### Output

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.

#### Ví dụ

Input	Output
2	B20DCCN002 Nguyen Quang Hai D20DCCN02-B 01/09/1994 3.00
ngUYen Van NaM D20DCCN01-B	B20DCCN001 Nguyen Van Nam D20DCCN01-B 02/12/1994 2.17

2/12/1994	
2.17	
Nguyen QuanG hAi	
D20DCCN02-B	
1/9/1994	
3.0	

Source : <https://ideone.com/NSyuSQ>

### Bài 20. Sắp xếp danh sách mặt hàng

Xây dựng cấp trúc mặt hàng gồm các thông tin: Mã mặt hàng (là một số nguyên, tự động tăng, tính từ 1); Tên mặt hàng, nhóm hàng: là các xâu ký tự; Giá mua, giá bán: là các số thực (không quá 9 chữ số)

Hãy sắp xếp danh sách các mặt hàng theo lợi nhuận giảm dần.

#### Input:

Dòng đầu chứa số mặt hàng. Mỗi mặt hàng viết trên 4 dòng: Dòng 1: Tên mặt hàng. Dòng 2: Nhóm hàng. Dòng 3: Giá mua. Dòng 4: Giá bán

**Output:** Ghi ra danh sách mặt hàng đã sắp xếp theo lợi nhuận giảm dần (lợi nhuận tính bằng giá bán trừ đi giá mua). Mỗi mặt hàng viết trên một dòng gồm: mã, tên, nhóm hàng và lợi nhuận. Các thông tin cách nhau đúng 1 khoảng trống. Lợi nhuận viết với 2 chữ số sau dấu phẩy.

#### Ví dụ:

Input	Output
3	2 Tu lanh Side by Side Dien lanh 7699.00
May tinh SONY VAIO	1 May tinh SONY VAIO Dien tu 1299.00
Dien tu	3 Banh Chocopie Tieu dung 9.50
16400	
17699	

Tu lanh Side by Side	
Dien lanh	
18300	
25999	
Banh Chocopie	
Tieu dung	
27.5	
37	

Source : <https://ideone.com/a6BJEY>