

به نام خداوند دانا و توانا

تمرین سری چهارم مبانی برنامه نویسی

مهلت ارسال: ۱۲ نیمه شب ۸۶/۳/۵

توضیحات :

۱. قبل از شروع موارد زیر را به دقت بخوانید.
۲. در خط اول هر یک از برنامه‌ها شماره دانشجویی خود را به فرمت زیر بنویسید:
// ID: your Student ID
مثال:
// ID: 87654321
۳. بعد از نوشتن همه برنامه‌ها ، همه آن‌ها را در یک فولدر با نام assignment#4_stdID قرار دهید) که در آن stdID شماره دانشجویی شما می باشد. مثلاً: assignment#4_87654321 و سپس این فولدر را فشرده کنید. فایل فشرده شده را به آدرس cpphomework852@gmail.com ارسال نمایید.
۴. عنوان ایمیل باید به صورت assignment#4_stdID باشد ، که در آن stdID شماره دانشجویی شما می باشد.
۵. اشتباه در نام‌گذاری فایل برنامه‌ها و فایل فشرده شده و یا عنوان ایمیل فرستاده شده ، موجب عدم تصحیح تمرین شما می‌شود.
۶. فرمت ورودی و خروجی برنامه‌های شما باید دقیقاً به صورت گفته شده در مسئله باشد ، در غیر این صورت چون برنامه‌های شما به صورت خودکار تصحیح می شود، خروجی برنامه‌ی شما با خروجی برنامه‌ی مصحح منطبق نخواهد بود.
۷. به هر یک از برنامه‌های شما تعدادی تست داده خواهد شد. توصیه می شود که حتماً یک بار تست‌های نمونه را به برنامه‌های تان بدهید.
۸. به برنامه‌های مشابه به میزان نمره سوال ، نمره منفی تعلق خواهد گرفت .
۹. موفق باشید.

نام فایل برنامه : Domino.cpp

Dominoes – game played with small, rectangular blocks of wood or other material, each identified by a number of dots, or pips, on its face. The blocks usually are called bones, dominoes, or pieces and sometimes men, stones, or even cards. The face of each piece is divided, by a line or ridge, into two squares, each of which is marked as would be a pair of dice...

The principle in nearly all modern dominoes games is to match one end of a piece to another that is identically or reciprocally numbered.

Given a set of domino pieces where each side is marked with two digits from 0 to 6. Your task is to arrange pieces in a line such way, that they touch through equal marked sides. It is possible to rotate pieces changing left and right side.

Input

The first line of the input contains a single integer N ($1 \leq N \leq 100$) representing the total number of pieces in the domino set. The following N lines describe pieces. Each piece is represented on a separate line in a form of two digits from 0 to 6 separated by a space.

Output

Write “No solution” if it is impossible to arrange them described way. If it is possible, write any of way. Pieces must be written in left-to-right order. Every of N lines must contains number of current domino piece and sign “+” or “-“ (first means that you not rotate that piece, and second if you rotate it).

Sample Input	Sample Output
5	2 -
1 2	5 +
2 4	1 +
2 4	3 +
6 4	4 -
2 1	

مسئله دوم

نام فایل برنامه : Database.cpp

در فایلی (input.txt) اطلاعات دانشجویان را ذخیره کرده‌ایم. اطلاعات شامل موارد زیر است:
شماره دانشجویی، نام، دوره، رشته، معدل، نام درس اخذ شده
شما باید با توجه به اطلاعات ذخیره شده، به سوالات زیر پاسخ دهید. از ورودی شماره سوال و سپس
ورودی‌های آن را دریافت کنید و سپس خروجی مناسب آن را چاپ نمایید.
سوالات:

۰. خروج از برنامه

۱. شماره‌ی دانشجویی دانشجویان یک رشته‌ی خاص را هر کدام در یک خط بنویسید.

ورودی: نام رشته

۲. نام دانشجویانی که معدل آنها بین ۲ عدد خاص است. هر کدام در یک خط بنویسید.

ورودی: ۲ عدد اعشاری به ترتیب صعودی

۳. نام و شماره‌ی دانشجویی دانشجویانی که درس خاصی را اخذ کرده‌اند. در هر خط ابتدا

نام سپس با یک فاصله شماره‌ی دانشجویی یک دانشجو را بنویسید.

ورودی: نام درس

۴. نام دانشجویان یک دوره تحصیلی که در رشته‌ی خاصی تحصیل می‌کنند. در هر خط

یک نام را بنویسید.

ورودی: دوره تحصیلی و سپس با یک فاصله نام رشته

فرمت فایل ورودی به صورت زیر است:

input.txt	ساختار فایل
student1 444 George Bush bs politics 10.02 Afghanistan Mathematics Iraq N.Korea	#دانشجو شماره دانشجویی نام دوره رشته معدل
Student2 777 Pesare Shoja bs computer 16.1 OS Compiler Circuits Network C++	درس ۱ درس ۲ درس ۳
student3 666 ??? bs electrical 19.98 Electronics Circuits C++ Mathematics	
student4 333 Pesar Amoo bs computer 14.2 Network Ecommerce ProjectManagement	

Sample input	Sample output
1 Computer 4 bs politics 3 Mathematics 0	Pesare Shoja Pesar Amoo George Bush George Bush ???

تمرین سری پنجم مبانی برنامه نویسی

مهلت ارسال: ۱۲ نیمه شب ۸۶/۳/۱۵

توضیحات :

۱. قبل از شروع موارد زیر را به دقت بخوانید.
 ۲. در خط اول هر یک از فایل‌ها شماره دانشجویی خود را به فرمت زیر بنویسید:
// ID: your Student ID
مثال:
// ID: 87654321
 ۳. بعد از نوشتن همه فایل‌ها ، همه آن‌ها را در یک فولدر با نام `assignment#5_stdID` قرار دهید) که در آن `stdID` شماره دانشجویی شما می باشد. مثلاً: `assignment#5_87654321` و سپس این فولدر را فشرده کنید. فایل فشرده شده را به آدرس cpphomework852@gmail.com ارسال نمایید.
 ۴. عنوان ایمیل باید به صورت `assignment#5_stdID` باشد ، که در آن `stdID` شماره دانشجویی شما می باشد.
 ۵. اشتباه در نام‌گذاری فایل برنامه‌ها و فایل فشرده شده و یا عنوان ایمیل فرستاده شده ، موجب عدم تصحیح تمرین شما می‌شود.
 ۶. برای تست برنامه‌ی شما، یک فایل شامل تابع `main` نوشته خواهد شد. فایل `map` شما در این فایل `include` شده و تمام توابع شما تست خواهد شد. بنابراین نیازی به ارسال فایل حاوی تابع `main` نیست.
 ۷. به هر یک از برنامه‌های شما تعدادی تست داده خواهد شد. توصیه می‌شود که حتماً یک بار تست‌های نمونه را به برنامه‌های تان بدهید.
 ۸. به برنامه‌های مشابه به میزان نمره سوال ، نمره منفی تعلق خواهد گرفت .
- 🚩 **نکته مهم:** پیاده‌سازی برنامه در دو فایل `map.h` و `map.cpp` الزامی است.
۹. موفق باشید.

نام فایل برنامه : **Map.h** و **Map.cpp**

Map یک داده ساختار است که دو عدد را به هم نگاشت می‌کند. عدد اول را کلید و عدد دوم را مقدار می‌نامند. در یک نمونه از map هیچ دو کلید مشابهی وجود ندارد اما ممکن است مقادیر تکراری باشند. می‌توانیم هر تعداد عنصر که بخواهیم به آن اضافه کنیم. در map ترتیب عناصر مهم نیست و تنها شرط عدم تشابه کلیدها لازم است. یک کلاس به نام map پیاده‌سازی کنید که بتوان با آن عملیات لازم را برای کنترل map انجام دهیم. constructor, copy constructor و destructor را پیاده‌سازی کنید. متدهای زیر را نیز پیاده‌سازی کنید:

```
Map();  
Map(Map &m);  
~Map();  
void insert(char *key, char *value);  
عنصر جدید را با کلید و مقدار داده شده در صورت امکان اضافه می‌کند.  
void insert(Map addThis);  
عناصر map ورودی را به ازای کلیدهای غیر مشابه به map فعلی اضافه می‌کند.  
void remove(char *key);  
اگر عنصری با کلید مورد نظر موجود باشد، آن را حذف می‌کند.  
int size();  
اندازه‌ی کل map را برمی‌گرداند.  
bool isEmpty();  
اگر map خالی باشد مقدار true و اگر نه false برمی‌گرداند.  
int getValue(char *key);  
مقدار متناظر با کلید را برمی‌گرداند.  
int getIndex(char *key);  
اندیس کلید مورد نظر را برمی‌گرداند.  
char** getKeys();  
آرایه‌ای از کلیدها را برمی‌گرداند.  
char** getValues();  
آرایه‌ای از مقادیر را برمی‌گرداند.
```

به برنامه‌هایی که کلاس را به صورت template پیاده‌سازی کنند (نوع کلید و مقدار)، نمره‌ی اضافی تعلق خواهد گرفت.