

کدوم سايز؟؟

در اين سوال قرار است ابتدا يک آرايه با اندازه y مشخص شده توسط کاربر ايجاد کنيم و سپس پس از انجام تغييراتي با دريافت دوباره از کاربر سايز آرايه را تغيير دهيم.

ورودی

به عنوان ورودی ابتدا عدد $n1$ به عنوان سايز اوليه آرايه داده می شود وظیفه y شما اين است که آرايه ای از integer ها به اندازه y بسازيد. سپس آرايه y مورد نظر را با مقدار 100 پر کنید.

پس از آن کاربر عدد $n2$ را وارد می کند که سايز جديد آرايه می باشد در صورتی که سايز جديد از سايز قبلی بزرگتر بود بايد خانه های جديد آرايه با مقادير 0 پر شوند.

خروجی

خانه های آرايه را پس از تغيير سايز دادن چاپ نماييد

نکته مهم:

دقت شود تنها بايد از يک آرايه استفاده کنید و نمی توانيد برای $n1$ يک آرايه و برای $n2$ آرايه y دیگری درست نماييد.

همچنين اين تمرين نیز به هيچ عنوان نمی توانيد از براکت [] استفاده کنید و بايد از ويژگی های پوينتر استفاده شود.

و حواستان باشد برنامه y شما memory leak نداشته باشد.

مثال

در اینجا چند نمونه برای فهم بهتر صورت سوال و قالب ورودی و خروجی تست ها داده می شود.

ورودی نمونه ۱

5 3

خروجی نمونه ۱

100 100 100

ورودی نمونه ۲

5 7

خروجی نمونه ۲

100 100 100 100 100 0 0

ضرب ماتریس ها

در این سوال به عنوان ورودی دو ماتریس داده می شود و به عنوان خروجی باید ضرب دو ماتریس را حساب کنید.

در سوال ابتدا سه عدد داده می شود عدد اول و دوم به ترتیب سطر و ستون ماتریس اول و عدد دوم و سوم نیز سطر و ستون ماتریس سوم می باشد.

سایز آرایه دوبعدی مشخص نیست و باید به صورت داینامیک حافظه گرفته شود.

در سوال باید توابع زیر را پیاده سازی نمایید.

```
void CreateMatrix( /*input*/ )
```

در این تابع به عنوان ورودی، ماتریس و تعداد سطر و ستونش را گرفته و حافظه ی لازم را به آن ماتریس تخصیص می دهید.

```
void FillMatrix( /*input*/ )
```

در این تابع به عنوان ورودی، ماتریس و تعداد سطر و ستونش را گرفته و سپس درون تابع از کاربر ورودی گرفته و ماتریس را پر می کنید.

```
void PrintMatrix( /*input*/)
```

در این تابع به عنوان ورودی، ماتریس و تعداد سطر و ستونش را گرفته و آن را چاپ نمایید.

این سه تا تابع را حتما باید پیاده سازی نمایید. براساس سوال نیز می توانید توابع دیگری نیز پیاده سازی کنید تا کد شما کوتاه تر شود. (تابعی مانند حساب کردن ضرب دو تا ماتریس و ...)

نکته مهم:

از آنجایی که داریم برای آرایه دو بعدی خود به صورت پویا حافظه می گیریم پس حتما حواستان به memory

leak باشد. همچنین این تمرین دستی هم تصحیح می شود و 100 گرفتن در تست کیس ها نشانه ی 100 گرفتن در این سوال نمی باشد و توابعی که می نویسد خوانده می شود.

راهنمایی: برای ورودی توابع به اینکه به صورت call by value ورودی بگیرین یا call by reference توجه نمایید.

ورودی

در خط اول سه عدد داده می شود که نشان دهنده ی سطر و ستون دو ماترس می باشد. در خطوط بعدی نیز مقادیر درون ماتریس ها داده می شود.

خروجی

حاصل ضرب دو ماتریس را به فرمتی که در پایین نشان داده شده چاپ نمایید.

مثال

ورودی نمونه ۱

```
2 3 4
2 1 4
5 4 1
1 2 3 5
5 4 3 1
6 3 2 1
```

خروجی نمونه ۱

```
31 20 17 15
31 29 29 30
```

با توجه به خط اول ماتریس اول یک ماتریس 2 در 3 می باشد و ماتریس دوم نیز یک ماتریس 3 در 4 می باشد. سپس در خطوط بعدی مقادیر دو ماتریس داده شده است. در خروجی نیز با توجه به ماتریس های اولیه مون

خروجی در یک ماتریس 2 در 4 حساب شده است.

ورودی نمونه ۲

3 3 3
1 2 3
4 5 6
7 8 9
9 8 7
6 5 4
3 2 1

خروجی نمونه ۲

30 24 18
84 69 54
138 114 90

فروشگاه کتاب

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

فرض کنید یک ردیف کتاب داریم که آنها را در یک قفسه از فروشگاه از چپ به راست چیده‌ایم. در این کتابفروشی برای گذاشتن و برداشتن کتاب‌ها نظم خاصی وجود دارد به این صورت که اگر بخواهیم یک کتابی را در این قفسه بگذاریم فقط می‌توانیم آن را سمت چپ یا سمت راست کتاب‌ها بگذاریم و برای برداشتن نیز فقط می‌توانیم از سمت چپ کتاب برداریم.

در اصل می‌توانیم سه عمل زیر را روی این ردیف کتاب انجام دهیم:

۱. عبارت $AddRight\ X$: در این عمل کتاب با نام X را به سمت راست کتاب‌ها اضافه می‌کنیم.
۲. عبارت $AddLeft\ X$: در این عمل کتاب با نام X را به سمت چپ کتاب‌ها اضافه می‌کنیم.
۳. عبارت $RemoveLeft$: در این عمل کتاب سمت چپ را از قفسه برمی‌داریم.

می‌خواهیم برنامه‌ای بنویسیم که ابتدا یک دسته کتاب را به عنوان ورودی گرفته و سپس تعدادی از عمل‌های بالا را روی آن انجام دهد و دسته کتاب نهایی را به عنوان خروجی چاپ کند.

برای پیاده‌سازی این برنامه لازم است از داده‌ساختار لیست پیوندی استفاده کنید.

لیست پیوندی (Linked list) ساختاری شامل دنبال‌های از عناصر است که هر عنصر دارای اشاره‌گری به عنصر بعدی در دنباله است.

برای پیاده‌سازی لیست پیوندی برای این مسئله یک `struct Book` تعریف می‌کنید که شامل یک `string` `Name` برای نگهداری نام کتاب و یک `Book* Next` برای اشاره به عنصر بعدی است.

سمت چپ‌ترین کتاب را اولین کتاب و راست‌ترین کتاب را آخرین کتاب در نظر می‌گیریم و همچنین کتاب بعدی هر کتاب را کتاب بلافاصله سمت راست آن در نظر می‌گیریم. دو اشاره‌گر مانند `Book* first` و `Book* last`

برای اشاره به کتاب اول و کتاب آخر نگهداری می‌کنیم. پس از هر عمل حذف یا اضافه در سمت چپ یا راست یکی از این دو اشاره‌گر باید به روزرسانی شوند.

```
struct Book
{
    string Name;
    Book *Next;
};
```

برای هر عمل اضافه کردن کتاب ، باید از دستور malloc , realloc یا calloc برای گرفتن حافظه برای Book جدید استفاده کنید و برای هر عمل حذف ، برای جلوگیری از نشت حافظه باید کتاب مورد نظر را با استفاده از دستور free از حافظه پاک کنید.

در ادامه لیست دستوراتی که به برنامه داده می‌شود و مفهوم آنها آمده است:

command	description
AddLeft BookName	با دیدن این عبارت، باید یک کتاب به ابتدای فروشگاه(سمت چپ) اضافه شود
AddRight BookName	با دیدن این عبارت، باید یک کتاب به انتهای فروشگاه (سمت راست) اضافه شود
DeleteLeft	با دیدن این عبارت، باید چپترین کتاب در فروشگاه را حذف کنید
Exit	با دیدن این کاراکتر، برنامه به پایان میرسد و ابتدا تعداد کتابهای داخل فروشگاه را چاپ کنید و سپس لیست کتابهای داخل فروشگاه را به ترتیب از چپ به راست چاپ کنید

ورودی

ابتدا یک عدد n در ورودی داده می‌شود که نشانگر تعداد کتابهای داخل فروشگاه در ابتدای کار است سپس n رشته به ترتیب چپ به راست که هر کدام نام یکی از کتابهاست. (نام هر کتاب رشته‌ای به طول حداکثر ۱۳۳

میباشد و از حروف کوچک و بزرگ انگلیسی و اعداد تشکیل شده است. (ممکن است در نام یک کتاب *space* نیز وجود داشته باشد.) سپس در هر مرحله یکی از دستورات بالا داده می‌شود.

تعداد دستوراتی که به برنامه داده می‌شود حداکثر 10^6 می‌باشد.

خروجی

در سطر اول تعداد کتاب‌های موجود در فروشگاه چاپ شود. در سطرهای بعدی در هر سطر نام یک کتاب از کتاب‌های موجود در فروشگاه چاپ شود. ترتیب چاپ کتاب‌ها از چپ به راست می‌باشد.

مثال

ورودی نمونه

```
3
Mathematics
General Physics 2
Advanced Programming
DeleteLeft
AddLeft
Kelile va Demne
AddRight
Boostane Hafez
Exit
```

خروجی نمونه

```
4
Kelile va Demne
General Physics 2
Advanced Programming
Boostane Hafez
```


ایلیا و مشکلات با اعداد کسری در زبان C

ایلیا قصد داشته که توی برنامه هایی که مینویسه، از اعداد کسری استفاده کنه، اما از اونجایی که کار با float و مخصوصا double به شدت گاهی اذیت میکنه، تصمیم گرفته از شما کمک بگیره تا بتونه راحت و بی دردسر کسر هارو نگه داره! در این سوال قصد داریم مکانیزم تعریف اعداد گویا را با کمک **structure** ها انجام دهیم.

میدونید که هر عدد گویا از یک صورت و یک مخرج تشکیل شده است. پس `struct`ی تعریف کنید که دو فیلد صحیح صورت و مخرج داشته باشد به عنوان مثال:

```
1 | struct Rational {  
2 |     int a;  
3 |     int b;  
4 | };
```

که تعریف کسر به شکل $\frac{a}{b}$ است.

تکته: ساختار داده شده، الزاما ساختار مورد استفاده ی شما نخواهد بود و ممکن است نیاز به تغییرات داشته باشد.

حال توابع زیر را برای محاسبه ی اعمال کسری باید پیاده سازی کنید:

```
1 | void getRational(Rational *i);
```

این تابع صورت و مخرج کسر را از ورودی دریافت کرده و در آرگومان ورودی ذخیره میکند.

```
1 | void print(Rational i);
```

این تابع، آرگومان ورودی را به شکل a/b در خروجی نمایش می دهد.

```
1 | void simplify(Rational *i);
```

این تابع کسر ورودی را به ساده ترین حالت ممکن تبدیل می کند.

```
1 | Rational add(Rational q1 , Rational q2);
```

این تابع دو کسر ورودی را با هم جمع کرده و به عنوان خروجی تابع بر میگرداند.

```
1 | Rational subtract(Rational q1 , Rational q2);
```

این تابع کسر دوم را از کسر اول کم کرده و آن را به عنوان خروجی تابع بر میگرداند.

```
1 | Rational multiply(Rational q1 , Rational q2);
```

این تابع دو کسر را در هم ضرب میکند و به عنوان خروجی تابع بر میگرداند.

```
1 | void reverse(Rational *i);
```

این تابع کسر ورودی را معکوس می کند.

```
1 | Rational divide(Rational q1 , Rational q2);
```

این تابع دو کسر را بر هم تقسیم می کند.

ورودی و خروجی

پیاده کردن main برنامه اختیاری و به عهده ی خودتان است و تکالیف به صورت دستی تصحیح خواهد شد، اما دقت کنید در main حداقل یکبار تمام توابع خود را تست کنید. نکته ی مهم در تصحیح این است که توابع پیاده سازی شده، بهینه و جامع باشد؛ طوری که پاسخگوی تمام تست کیس ها باشد.

*تذکره:

- صورت و مخرج کسر ها اعداد صحیح خواهد بود.

- به Call by value یا Call by reference بودن توابع دقت کنید.
- تعریف توابع بالا اجباریه، اما میتونین خودتون برای راحتی کار توابع دیگه ای هم تعریف کنید که کارتون راحت باشه، یادتون باشه اینجا نمیخوایم کدمون فقط کار کنه، میخوایم بهینه کد بزنینم، پس حواستون به حجم کدتون باشه. (مثلا تعریف یک تابع ک.م.م حجم کدتون رو خیلی کمتر میکنه)
- سعی در کنترل دستی خطاها با شرط داشته باشید، حالات استثنا و غیر ممکن رو در نظر داشته باشید:

به نمونه پیاده سازی Main زیر دقت کنید:

```
1  int main(){
2  Rational Q1 , Q2 , Q3;
3  getRational(&Q1);
4  // فرض کنید کاربر عدد 2 و 5 را وارد میکند
5  getRational(&Q2);
6  // فرض کنید کاربر عدد 6 و 3 را وارد میکند
7
8  Q3 = add(Q1 , Q2);
9  print(Q3);
10 // باید مقدار 36/15 پرینت شود
11 simplify(&Q3);
12 print(Q3);
13 // باید مقدار 12/5 پرینت شود
14 }
```

برقی در کمیته انضباطی(اختیاری)

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

محمد مهدی به تازگی رئیس کمیته انضباطی فدراسیون فوتبال شده و می‌خواهد قبل از شروع لیگ تخلفات تیم‌های متخلف را شناسایی کرده و از شرکت آنها در لیگ ممانعت به عمل آورد! لیگ تا ۲ هفته دیگر شروع می‌شود و مهلت نقل و انتقالات به تازگی به پایان رسیده. اسامی بازیکنان مجاز جهت شرکت در لیگ به همراه اسامی تیم‌ها و بازیکنان هر تیم به سازمان لیگ داده شده است.

محمد مهدی که به تنهایی نمی‌تواند در بین همه اسامی تخلفات را پیدا کند از شما می‌خواهد که برنامه‌ای بنویسید که تخلفات تیم‌های مختلف را شناسایی کرده و گزارش کنید.

موارد زیر در صورت وقوع به عنوان تخلف شناخته خواهند شد:

- در بین اسامی بازیکنان یک تیم بازیکنی وجود داشته باشد که نام آن در لیست بازیکنان مجاز جهت شرکت در لیگ نیست. ← در این صورت آن تیم به عنوان متخلف شناخته خواهد شد.
- اسم یک بازیکن در چند تیم وجود داشته باشد (بازیکن با چند تیم قرارداد بسته باشد) ← در این صورت تمامی تیم‌هایی که بازیکن با آنها قرارداد بسته تخلف کرده اند.

ورودی:

در خط اول دو عدد n و m داده می‌شوند که به ترتیب تعداد بازیکن‌ها و تعداد تیم‌ها هستند.

$$1 \leq n \leq 5000$$

$$1 \leq m \leq 62$$

در n خط بعدی در هر خط نام یک بازیکن داده می‌شود که به صورت یک `string` و شامل حروف کوچک انگلیسی و فاصله (space) است.

دقت کنید که ممکن است نام بازیکن تکراری در ورودی داده شود.

سپس در خطوط بعدی به ازای تیم i ام در خط اول نام تیم i ام داده می‌شود که شامل حروف کوچک انگلیسی و space است و خط بعدی عدد ai که تعداد بازیکنان تیم i ام است داده می‌شود. سپس در ai خط بعدی بازیکنان تیم i ام داده می‌شود. $\$ \$ ai \leq 100 \geq 1 \$ \$$

خروجی:

در خروجی باید نام تیم‌های متخلف را به ترتیب حروف الفبا چاپ کنید.

مثال:

ورودی نمونه:

```
136 6
dani carvajal
eder militao
sergio ramos
raphael varane
nacho
eden hazard
toni kroos
karim benzema
luka modric
gareth bale
marcelo
thibaut courtois
casemiro
federico valverde
james rodriguez
lucas vazquez
luka jovic
marco asensio
brahim diaz
isco
ferland mendy
mariano
```

vinicius Junior
rodrygo
marc andre ter stegen
nelson semedo
gerard pique
ivan rakitic
sergio busquets
arthur
luis suarez
lionel messi
ousmane Dembele
neto
clement lenglet
antoine griezmann
jordi alba
sergi roberto
frenkie de jong
arturo vidal
samuel umtiti
junior firpo
tomas vaclik
sergi Gomez
lucas ocampas
daniel carrico
rony lopes
nolito
ever banega
munir
jules kounde
bono
suso
youssef en nesryi
jesus navas
nemanja gudelj
sergio escudero
luuk de jong
diego carlos
oliver torres
franco Vazquez
sergio reguilon
joan jordan

fernando
javi diaz
antonio adan
jose gimenez
santiago arias
thomas partey
koke
aoao felix
saul
alvaro morata
angel correa
thomas lemar
renan lodi
jan oblak
marcos llorente
stefan savic
hector herrera
ivan saponjic
felipe
diego costa
vitolo
yannick carrasco
mario hermoso
kieran trippier
sime vrsaljko
alex remiro
joseba zaldua
diego llorente
asier illarramendi
igor zubeldia
aritz elustondo
portu
mikel merino
willian jose
mikel oyarzabal
adnan januzaj
aihen munoz
miguel angel moya
ander guevara
david zurutuza
andoni gorosabel

alexander isak
nacho monreal
martin odegard
ander barrenetxea
luca sangalli
robin le normand
andoni zubiaurre
jaume domenech
thierry correia
jaume costa
eliaquim mangala
gabriel paulista
geoffrey kondogbia
goncalo guedes
carlos soler
kevin gameiro
dani parejo
denis cheryshev
mouctar diakhabay
jasper cillessen
jose gaya
manu vallejo
lee kang in
francis coquelin
daniel wass
rodrigo
ferran torres
cristiano piccini
maxi gomez
ruben sobrinho
ezequiel garay
alessandro florenzi
real madrid
25
dani carvajal
eder militao
sergio ramos
raphael varane
nacho
eden hazard
toni kroos

karim benzema
luka modric
gareth bale
marcelo
thibaut courtois
casemiro
federico valverde
james rodriguez
lucas vazquez
luka jovic
marco asensio
brahim diaz
martin odegaard
isco
ferland mendy
mariano
vinicius Junior
rodrygo
barcelona
18
marc andre ter stegen
nelson semedo
gerard pique
ivan rakitic
sergio busquets
arthur
luis suarez
lionel messi
ousmane Dembele
neto
clement lenglet
antoine griezmann
jordi alba
sergi roberto
frenkie de jong
arturo vidal
samuel umtiti
junior firpo
sevilla
23
tomas vaclik

sergi Gomez
lucas ocampo
daniel carrico
rony lopes
nolito
ever banega
munir
jules kounde
bono
suso
youssef en nesryi
jesus navas
nemanja gudelj
sergio escudero
luuk de jong
diego carlos
oliver torres
franco Vazquez
sergio reguilon
joan jordan
fernando
javi diaz
atletico madrid
23
antonio adan
jose gimenez
santiago arias
thomas partey
koke
aoao felix
saul
alvaro morata
angel correa
thomas lemar
renan lodi
jan oblak
marcos llorente
stefan savic
hector herrera
ivan saponjic
felipe

diego costa
vitolo
yannick carrasco
mario hermoso
kieran trippier
sime vrsaljko
real sociedad
23
alex remiro
joseba zaldua
diego llorente
asier illarramendi
igor zubeldia
aritz elustondo
portu
mikel merino
willian jose
mikel oyarzabal
adnan januzaj
aihen munoz
miguel angel moya
ander guevara
david zurutuza
andoni gorosabel
alexander isak
nacho monreal
martin odegaard
ander barrenetxea
luca sangalli
robin le normand
andoni zubiaurre
valencia
25
jaume domenech
thierry correia
jaume costa
eliaquim mangala
gabriel paulista
geoffrey kondogbia
goncalo guedes
carlos soler

kevin gameiro
dani parejo
denis cheryshev
mouctar diakhabay
jasper cillessen
jose gaya
manu vallejo
lee kang in
francis coquelin
daniel wass
rodrigo
ferran torres
cristiano piccini
maxi gomez
ruben sobrinho
ezequiel garay
alessandro florenzi

خروجی نمونه:

real madrid
real sociedad

نام بازیکن martin odegaard هم در تیم real madrid وجود دارد هم در تیم real sociedad. بنابراین هر دو تیم متخلف محسوب شده و نام آنها به ترتیب حروف الفبا نمایش داده می‌شود.