

## نقشه پلاستیکی

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

برای آشنایی با مشکلات یک نقشه و دست گرمی مشکل ساده شده ایی از تولید و رسم به نقشه در اسنپ به شما داده شده است. در ابتدا ما فقط مختصات دقیق مکان start را داریم. سپس تمام نقاط دیگر به صورت یک زنجیره از این نقطه به ما داده می‌شود. حال وظیفه شما پیدا کردن مختصات دقیق هر مکان می‌باشد.

## ورودی

در خط اول مختصات start می‌آید. مقدار x و y که از نوع integer هستند در ابتدا داده می‌شود.

در خطوط بعدی، در هرخط یک مکان جدید معرفی می‌شود. در ابتدا نام مکان جدید سپس عبارت from و بعد از آن نام مکانی که x و y نسبت به آن داده می‌شود. همواره در این خطوط مقدار x و y با علامت + یا - داده می‌شود.

مثلا در مثال اول hospital مقدار x آن یک واحد کمتر از x متعلق به start است و مقدار y آن دو واحد بیشتر از y متعلق به start است.

## خروجی

به ازای هر مکان از ورودی به جز start در خروجی باید مختصات دقیق همان مکان، با رعایت ترتیبی که در ورودی آمده اند چاپ شود.

## مثال

### ورودی نمونه ۱

```
start x=1 y=1
hospital from start x=-1 y=2
```

### خروجی نمونه ۱

```
hospital x=0 y=3
```

### ورودی نمونه ۲

```
start x=1 y=1
office from hospital x=+4 y=-1
hospital from start x=-1 y=+2
```

### خروجی نمونه ۲

```
office x=4 y=2
hospital x=0 y=3
```

## نکات بیشتر

- ترتیب نمایش مکان ها در خروجی باید دقیقا به ترتیب ورودی باشد(به جز start که نیازی به نمایش آن نیست).
- تضمین میشود راه حل وجود دارد.
- تضمین میشود نام هر مکان یکتا و تشکیل شده از حروف کوچک انگلیسی و اعداد است و بعد از خط اول یک مکان جدید توصیف شود.
- تضمین میشود تعداد کل مکان ها بصورت زیر باشد.

$$1 \leq n \leq 1000$$

- مقدار نهایی x و y بصورت زیر باشد.

$$-1000000 \leq x,y \leq 1000000$$

## آن‌چه باید آ‌پلود کنید

یک فایل main.go که شامل پیاده‌سازی کامل راه حل باشد.

## رمزنگاری شبکه

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

بعد از اینکه قلی کالج گولنگ کوئرا را به اتمام رساند حالا به دنبال استخدام در شرکت‌های بزرگ است، و تصمیم گرفته برای یک موقعیت برنامه‌نویسی در شرکت اسنپ درخواست دهد. شرکت اسنپ برای ارزیابی مهارت‌های برنامه‌نویسی قلی، به او یک مسئله امنیتی داده تا مطمئن شود او توانایی لازم برای پیوستن به تیم را دارد. قلی باید یک پروتکل ارتباطی امن طراحی کند که برای ارسال پیام‌ها در شبکه‌ی داخلی شرکت اسنپ استفاده شود. این پروتکل باید به گونه‌ای باشد که پیام‌ها را به طور کامل رمزنگاری کند.

قوانین رمزنگاری به شرح زیر است:

### ۱. شیفت حروف و Emoticons

- هر حرف یا ایموجی در پیام باید به اندازه‌ی باقی‌مانده‌ی تقسیم *Unicode* آن حرف بر ۵، در الفبا به جلو شیفت داده شود.
- اگر شیفت به انتهای الفبا یا بازه Emoticons برسد، به ابتدای آن ادامه یابد.

### ۲. جایگزینی اعداد

- هر عدد در پیام باید با کد *Unicode* خود جایگزین شده و سپس در طول پیام همان ضرب شود.
- برای مثال، اگر عدد ۳ در پیام باشد و طول پیام قبل از رمزنگاری ۱۰ باشد، عدد ۳ به کد *Unicode* خود (51) جایگزین شده و سپس با ۱۰ ضرب می‌شود، نتیجه می‌شود 510.
- طول پیام یعنی تعداد character های آن.

### ۳. تبدیل فضا‌های خالی

- هر فضای خالی باید به تعداد character های کلمه قبل از آن به آندرلاین \_ تبدیل شود.
- برای مثال، اگر قبل از فضای خالی، کلمه‌ای با ۵ حرف باشد، آن فضای خالی به \_\_\_\_\_ تبدیل می‌شود.

### ۴. معکوس کردن پیام

- پیام نهایی پس از اعمال تمامی قوانین بالا باید به صورت معکوس چاپ شود (به جز اعداد).

**نکته:** این موارد به همین ترتیب روی هر پیام اعمال می‌شوند.

قلی نیاز دارد که برنامه‌ای بنویسد که ورودی یک پیام را دریافت کرده و آن را مطابق با قوانین بالا رمزنگاری کند و نتیجه را چاپ کند.

## نگته

- برای درک بهتر مفاهیمی مثل rune و character خواندن این دو مقاله (لینک، لینک) از نان شب واجب‌تر است!!
- تضمین می شود فقط حروف بزرگ و کوچک انگلیسی و اعداد و فاصله و ایموجی های emoticons در عبارات باشد.
- برای بهتر متوجه شدن emoticons و بازه مورد نیاز در unicode توصیه می شود این لینک خوانده شود.

## ورودی

- ابتدا یک عدد صحیح n دریافت می‌کنید که نشان‌دهنده تعداد جملاتی‌ست که قرار است برای رمزنگاری از ورودی دریافت کنید.
- در n خط بعدی ورودی یک جمله می‌آید که می‌تواند شامل حروف کوچک و بزرگ، اعداد و فضا‌های خالی باشد.

## خروجی

- در هر خط خروجی رشته‌ای که پیام رمزنگاری‌شده نهایی را نمایش می‌دهد چاپ کنید.

## مثال

### ورودی نمونه ۱

```
1
Hello Wor1d 123
```

### خروجی نمونه ۱

```
765750735____dovpY____poofJ
```

### ورودی نمونه ۲

```
4
cCT6gxy
pzZL7A2RJQ1knB8BCEAz 🍌
OAtv wTVpq0HndG kHs1
12JQA1Gsu
```

### خروجی نمونه ۲

```
zxj378Xeg
🍌bAIEC1176Cnm1RNT1050A1155MZbr
980sJm_____HdnJ960trwXa____yuAS
wsH1ARN450o
```

## آنچه باید آپلود کنید

این سوال در قالب ورودی و خروجی است، بنابراین یک فایل `main.go` آپلود کنید که در آن ورودی را خوانده و خروجی را به شکل خواسته شده چاپ می‌کند.

## فروشگاه پوشاک

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

گلناز خواهر قلی دارای یک فروشگاه پوشاک است که تصمیم گرفته برای تشویق مشتریان به خرید چندین لباس از مجموعه‌های مختلف خود، تخفیف‌های ویژه‌ای برای خریدهای چندتایی در نظر بگیرد. از آنجایی که برادرش قلی به تازگی وارد کالج گولنگ کوئرا شده است از برادرش میخواهد کدی برای او بنویسد که قیمت نهایی سبد خرید مشتری را با بیشترین تخفیف ممکن محاسبه کند.

فروشگاه گلناز ۵ نوع پوشاک مختلف دارد:

- پیراهن (shirt)
- شلوار (pants)
- کاپشن (jacket)
- کفش (shoes)
- هودی (hoody)

قیمت هر لباس در حالت عادی ۸۰۰ هزار تومان (۸۰۰۰۰۰۰۰ ریال) است، اما اگر چند نوع مختلف از این پوشاک را با هم خریداری کنید، تخفیف‌های زیر اعمال می‌شود:

- اگر ۲ نوع لباس مختلف بخرید، ٪۵ تخفیف می‌گیرید.
- اگر ۳ نوع لباس مختلف بخرید، ٪۱۰ تخفیف می‌گیرید.
- اگر ۴ نوع لباس مختلف بخرید، ٪۲۰ تخفیف می‌گیرید.
- اگر هر ۵ نوع لباس را بخرید، ٪۲۵ تخفیف می‌گیرید.

**نکته:** اگر اقلام خریداری شده شامل نسخه‌های تکراری از یک نوع لباس باشد، تخفیف فقط برای تعداد لباس‌های متفاوت اعمال می‌شود. به عنوان مثال، اگر ۴ لباس بخرید که ۳ نوع آنها متفاوت باشد، ٪۱۰ تخفیف برای آن ۳ نوع لباس لحاظ می‌شود و لباس چهارم همچنان با قیمت کامل ۸۰۰ هزار تومان محاسبه می‌شود.

**نکات:** تضمین می‌شود نهایت ۱۵ عدد لباس در هر سبد است.

## ورودی

- در ورودی یک آرایه از لباس‌ها به شما داده می‌شود برای مثال ورودی

shirt, shirt, pants, pants, jacket, jacket, shoes, hoody

نشان‌دهنده خرید دو عدد پیراهن، دو عدد شلوار، دو عدد کاپشن، یک جفت کفش و یک عدد هودی است.

## خروجی

- باید یک عدد صحیح چاپ کنید که هزینه کل سبد خرید را بر حسب ریال پس از اعمال تخفیف‌ها نشان دهد.

## مثال

## ورودی نمونه ۱

shirt, shirt, pants, pants, jacket, jacket, shoes, hoody

## خروجی نمونه ۱

51200000

دو روش برای گروه‌بندی این ۸ لباس وجود دارد:

**روش اول** تشکیل یک گروه ۵ تایی و یک گروه ۳ تایی:

- گروه اول شامل ۵ نوع لباس مختلف (پیراهن، شلوار، کاپشن، کفش، هودی) می‌باشد که هزینه آن با ٪۲۵ تخفیف برابر با سه میلیون تومان محاسبه می‌شود.
- گروه دوم شامل ۳ نوع لباس مختلف باقی مانده (پیراهن، شلوار، کاپشن) می‌باشد که هزینه آن با اعمال ٪۱۰ تخفیف برابر با دو میلیون و ۱۶۰ هزار تومان محاسبه می‌شود.

قیمت نهایی در این روش برابر با ۵ میلیون و ۱۶۰ هزار تومان یا ۵۱۶۰۰۰۰۰ ریال می‌شود.

**روش دوم** تشکیل دو گروه ۴ تایی:

- گروه اول شامل ۴ نوع لباس مختلف ( پیراهن، شلوار، کاپشن، کفش) بوده که هزینه آن با ۲۰ درصد تخفیف برابر با دو میلیون و ۵۶۰ هزار تومان محاسبه می‌شود.
- گروه دوم شامل ۴ نوع لباس مختلف باقی‌مانده (پیراهن، شلوار، کاپشن، هودی) که هزینه آن نیز با ٪۲۰ تخفیف برابر با دو میلیون و ۵۶۰ هزار تومان محاسبه می‌شود.

قیمت نهایی در این روش برابر با ۵ میلیون ۱۲۰ هزار تومان یا ۵۱۲۰۰۰۰۰ ریال می‌شود.

پاسخ صحیح ۵۱۲۰۰۰۰۰ ریال است، چون تخفیف بیشتری در این حالت اعمال می‌شود.

## ورودی نمونه ۲

shirt,pants,shoes,pants,shirt,shirt,jacket,jacket,hoody

خروجي نمونه ۲

59200000

این نمونه میتواند شامل چندین روش گروهبندی با لباس‌های مختلف باشد (جزئیات گروهبندی به دلیل تنوع بالا آورده نشده است):

روش اول: یک گروه ۵ تایی با ۲۵ درصد تخفیف، یک گروه ۳ تایی با ۱۰ درصد تخفیف و یک نوع لباس بدون تخفیف با هزینه ۵۹۶۰۰۰۰۰ ریال.

روش دوم: دو گروه ۴ تایی با ۲۰ درصد تخفیف و یک لباس تکی بدون تخفیف با هزینه ۵۹۲۰۰۰۰۰ ریال.

روش سوم: سه گروه ۳ تایی با تخفیف ۱۰ درصد با هزینه ۶۴۸۰۰۰۰۰ ریال.

روش چهارم: یک گروه ۴ تایی با تخفیف ۲۰ درصد و یک گروه ۳ تایی با تخفیف ۱۰ درصد و یک گروه ۲ تایی با تخفیف ۵ درصد با هزینه ۶۳۴۰۰۰۰۰ ریال.

روش پنجم: یک گروه ۵ تایی با ۲۵ درصد تخفیف و دو گروه ۲ تایی با ۵ درصد تخفیف با هزینه ۶۰۴۰۰۰۰۰ ریال.

پاسخ صحیح با توجه به مقادیر بالا مقدار ۵۹۲۰۰۰۰۰ است که بیشترین تخفیف را داراست.

آنچه باید آپلود کنید

یک فایل main.go که شامل پیاده‌سازی کامل راه حل باشد.

## تاریخچه‌ی مرورگر قلی

- محدودیت زمان: ۰.۵ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

از وقتی که قلی وارد اسنپ شد به خودش قول داد که هر روز به تمرین الگوریتم و ساختار داده بپردازد. اینبار قلی می‌خواهد با استفاده از linked list تاریخچه مرورگر خود را شبیه‌سازی کند.

وظایف این برنامه به شرح زیر است:

۱. سیستم باید بتواند صفحات وب جدیدی را که قلی بازدید می‌کند به تاریخچه اضافه کند.
۲. سیستم باید اجازه دهد که قلی به صفحه قبلی (Backward) یا صفحه بعدی (Forward) در تاریخچه مرورگر خود حرکت کند.
۳. هر زمان که قلی یک صفحه جدید بازدید می‌کند، اگر در وسط تاریخچه باشد (یعنی در آخرین صفحه بازدید شده نباشد)، باید تمام صفحات جلویی را حذف کرده و صفحه جدید را به انتهای تاریخچه اضافه کند.
۴. سیستم باید بتواند تمام تاریخچه مرورگر را پاک کند و به حالت اولیه بازگرداند.
۵. قلی باید بتواند URL صفحه فعلی که در حال مشاهده آن است را دریافت کند.

### آن‌چه باید پیاده‌سازی کنید

پروژه‌ی اولیه را از این لینک دانلود کنید. ساختار فایل‌های پروژه به صورت زیر است:

```
.
├── go.mod
├── go.sum
├── main.go
└── main_sample_test.go
```

ساختارها و متدهایی که باید پیاده‌سازی کنید:

#### تایپ BrowserHistory

شما باید یک ساختار BrowserHistory ایجاد کنید که تمامی داده‌های لازم برای مدیریت تاریخچه مرورگر را نگهداری کند. می‌توانید هر فیلد دیگری به این استراکت اضافه کنید.

#### تابع NewBrowserHistory

این تابع یک نمونه از BrowserHistory را مقداردهی اولیه می‌کند و برمی‌گرداند.

#### متد VisitNewPage

این متد باید یک صفحه جدید به تاریخچه اضافه کند. اگر کاربر در وسط تاریخچه قرار دارد، باید تمام صفحات جلویی را حذف کرده و صفحه جدید را به انتهای تاریخچه اضافه کند.

#### متد Back

این متد باید کاربر را به صفحه قبلی در تاریخچه منتقل کند. اگر در ابتدای تاریخچه باشیم، این متد باید ارور no previous page را برگرداند.

#### متد Forward

این متد باید کاربر را به صفحه بعدی در تاریخچه منتقل کند. اگر در انتهای تاریخچه باشیم، این متد باید ارور no next page را برگرداند.

#### متد ClearHistory

این متد باید تمام تاریخچه مرورگر را پاک کرده و سیستم را به حالت اولیه بازنشانی کند.

#### متد GetCurrentURL

این متد باید URL صفحه فعلی که کاربر در حال مشاهده آن است را برگرداند.

#### نگات بیشتر

- برای پیاده‌سازی این سیستم، از لیست پیوندی دوطرفه استفاده کنید و مطمئن شوید که تمامی عملکردها به صورت بهینه پیاده‌سازی شده‌اند.
- هیچکدام از ساختارهای اولیه پروژه را تغییر ندهید (امضای توابع و فیلدهای استراکت‌ها)، در این صورت تست‌ها پاس نخواهند شد.
- می‌توانید از تست‌های فایل main\_sample\_test.go استفاده کنید. توصیه می‌کنیم به همین روش برای خود تست‌های بیشتری بنویسید و برنامه خود را در حالت‌های مختلف تست کنید.

### آن‌چه باید آپلود کنید

یک فایل main.go که شامل پیاده‌سازی کامل ساختارها و توابع ذکر شده باشد.

## اسنپ اکسپرس

- محدودیت زمان: ۵ ثانیه
- محدودیت حافظه: ۱۰۲۴ مگابایت

همه‌ی دوستان و آشنایان قلی اتفاق نظر دارند که او از وقتی وارد اسنپ شده جواب دیگران را نمی‌دهد و فکر می‌کند بهترین برنامه‌نویس دنیااست! مهندس ارشد قلی برای اینکه کمی حال او را بگیرد تصمیم گرفت یک مسئله‌ی خیلی سخت درباره‌ی یکی از مشکلات داخلی اسنپ به او بدهد تا حساب کار دستش بیاید، قلی هم باز برای حل کردن آن دست به دامن شما شده.

## مسئله

اسنپ می‌خواهد رستوران‌هایی را که سفارش‌هایشان را با تاخیر ارسال می‌کنند پیدا کند. بنابراین از قلی می‌خواهند که برنامه‌ای بنویسد تا بتواند از روی لیست سفارش‌های هر روز این مورد را تشخیص دهد. مهندسان اسنپ همچنین به منظور اسکیل کردن برنامه‌هایشان می‌خواهند بدانند که در هر روز چه تعداد سفارش فعال در اپلیکیشن وجود دارد. برنامه قلی باید بتواند این دو کار را انجام دهد.

## ورودی

در سطر اول عدد صحیح  $n$  می‌آید که تعداد رستوران‌ها را مشخص می‌کند.

$$10 \leq n \leq 100$$

در  $n$  سطر بعدی هر خط شامل این موارد به شکل زیر است: نام یک رستوران و میانگین زمانی است که این رستوران برای تکمیل سفارش‌ها نیاز دارد به دقیقه. این عدد از روی سفارشات قبلی رستوران در سیستم محاسبه می‌شود.

```
kababi haj ali 30
sushi express 45
...
```

در سطر بعدی عدد صحیح  $m$  که تعداد سفارشات آن روز را مشخص می‌کند.

$$100 \leq m \leq 1000$$

در  $m$  خط بعدی هر خط به شکل زیر شامل اطلاعات یک سفارش است: دو اسم که در ابتدا و انتهای سطر می‌آیند و دو زمان.

```
kababi haj ali 10:25 11:15 Gholi
Zahra 15:17 14:50 feri kasif sandwich
```

همانطور که در مثال بالا مشاهده می‌کنید، تاریخچه سفارشات یک دست و منظم نیستند پس قلی برای نوشتن برنامه باید حواسش را خیلی جمع کند. قوانین لیست سفارشات به این صورت است:

- اگر سطر با نام یک رستوران شروع شود (که سیستم باید به طور خودکار تشخیص دهد)، زمان اول، زمان ثبت سفارش توسط مشتری و زمان دوم زمان دریافت سفارش توسط مشتری است.
- اگر خط با نام یک مشتری شروع شود، زمان اول زمان دریافت سفارش توسط مشتری و زمان دوم زمان تایید سفارش توسط رستوران است.
- وقتی یک سفارش در اپلیکیشن ثبت شود به طور متوسط ۵ دقیقه طول می‌کشد تا رستوران سفارش را دیده و تایید کند.
- وقتی غذا آماده می‌شود به طور متوسط ۱۰ دقیقه طول می‌کشد تا توسط پیک به مشتری برسد.

به عنوان نمونه در مثال قبل:

قلی ساعت ۱۰:۲۵ سفارش خود را ثبت می‌کند. پس کبابی حاج علی ساعت ۱۰:۳۰ سفارش را تایید خواهد کرد. از طرفی قلی در ساعت ۱۱:۱۵ سفارش خود را تحویل می‌گیرد. پس رستوران در ساعت ۱۱:۰۵ غذا را آماده کرده بوده. در نتیجه زمان آماده‌سازی غذا توسط کبابی حاج علی در این مورد ۳۵ دقیقه است. با توجه به اینکه آماده‌سازی غذا در این رستوران به طور میانگین ۳۰ دقیقه طول می‌کشد، پس این سفارش با تاخیر ارسال شده است.

و در نهایت در سطر آخر یک کلمه می‌آید که یا نام یک رستوران است یا عبارت `overlap`.

- اگر نام رستوران داده شود، در خروجی برنامه باید تعداد سفارشات تکمیل شده دیر هنگام آن رستوران را چاپ کنید.
- اگر `overlap` داده شود، باید ماکزیمم تعداد سفارشات همزمان را چاپ کنید.

## خروجی

- برای حالت نام رستوران: یک عدد صحیح، تعداد سفارشات آن روز برای آن رستوران که دیر تکمیل شده‌اند را چاپ کنید.
- برای حالت `overlap`: یک عدد صحیح، ماکزیمم تعداد سفارشاتی را که زمان‌های آن‌ها تداخل دارند در آن روز چاپ کنید.

## مثال

### ورودی نمونه ۱

```
2
kababi haj ali 30
sushi express 45
2
kababi haj ali 10:25 11:15 Gholi
Zahra 14:17 14:50 sushi express
kababi haj ali
```

### خروجی نمونه ۱

1

ورودی نمونه ۲

3  
burger king 25  
pizza hut 30  
taco bell 20  
4  
burger king 12:00 12:45 Alice  
Bob 13:40 13:10 pizza hut  
taco bell 11:50 12:30 Charlie  
Diana 14:35 14:00 burger king  
overlap

خروجی نمونه ۲

2

نکات

- برای پیاده‌سازی این سوال سعی کنید با امکانات مختلف گو آشنا شوید تا کار برای شما راحت‌تر باشد. به عنوان نمونه پکیج `time` می‌تواند امکانات خوبی برای پارس کردن زمان، اضافه و کم کردن چند دقیقه به زمان و غیره در اختیار شما بگذارد.
  - همچنین با کمک عبارات منظم *Regular Expressions* نیز می‌توانید اطلاعات معنی‌دار از رشته‌ها استخراج کنید.
- تداخل زمان‌ها: برای محاسبه تداخل‌ها، صورت مسئله را خوب بخوانید تا دقیقاً متوجه شوید که چه چیزی از شما می‌خواهد. برای مثال اگر در یک لحظه از روز ۳ سفارش فعال وجود داشته باشد و لحظه‌ای دیگر ۵ سفارش فعال، باید عدد ۵ در خروجی چاپ شود.
- ممکن است زمان تحویل بعضی سفارش‌ها از نیمه‌شب عبور کند. مثلاً سفارش می‌تواند در ساعت 23:55 توسط رستوران تایید شود و ساعت ۰۰:۲۵ به مشتری برسد. پس در محاسبه زمان‌ها دقت کنید!
  - در تمام محاسبات، زمان‌ها را به واحد یکسان (دقیقه) استفاده کنید تا از بروز اشتباهات جلوگیری شود.

آن‌چه باید آپلود کنید

این سوال در قالب ورودی و خروجی است، بنابراین یک فایل `main.go` آپلود کنید که در آن ورودی را خوانده و خروجی را به شکل خواسته شده چاپ می‌کند.