

کتابخانه

• محدودیت زمان: ۱ ثانیه

• محدودیت حافظه: ۲۵۶ مگابایت

فرض کنید یک ردیف کتاب داریم که آنها را در یک قفسه از کتابخانه از چپ به راست چیده‌ایم. در این کتابخانه 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 دنباله‌های 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 `new` 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 `delete` 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 کتابخانه (سمت چپ) اضافه شود |
| AddRight BookName | با دیدن این عبارت، باید یک کتاب به انتهای کتابخانه (سمت راست) اضافه شود |
| DeleteLeft | با دیدن این عبارت، باید چپترین کتاب در کتابخانه را حذف کنید |
| Exit | با دیدن این کاراکتر، برنامه به پایان میرسد و ابتدا تعداد کتابهای داخل کتابخانه را چاپ کنید و سپس لیست کتابهای داخل کتابخانه را به ترتیب از چپ به راست چاپ کنید |

ورودی

ابتدا یک عدد n در ورودی داده می‌شود که نشانگر تعداد کتابهای داخل کتابخانه 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 کتابخانه چاپ شود. در سطرهای بعدی در هر سطر نام یک کتاب از کتاب‌های موجود در کتابخانه 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
```

زنجیر مرتب

از آنجایی که آخر ترم است و نه شما وقت قصه خوندن دارید و نه من وقت نوشتن (-_-) بریم سر اصل مطلب:
یک لینکدلیست بنویسید که ویژگی‌های زیر را داشته باشد.

- یک عدد صحیح به عنوان age ذخیره کند.
- یک عدد اعشاری به عنوان وزن ذخیره کند.
- یک رشته به عنوان name ذخیره کند. (حداکثر ۲۰).

باید توابع زیر را برای این لیست پیاده‌سازی کنید.

- تابع insert : محتویات یک خانه و جای ذخیره سازی آن را از ورودی دریافت کند و آن را ذخیره کند.
- تابع delete : جای ذخیره سازی یک خانه را دریافت کند و اگر وجود داشت آن را حذف کند.
- تابع sort : از یکی از الگوریتم‌های منطقی مرتب‌سازی استفاده کنید و لینکدلیست را مرتب کنید و چاپ کنید. مرتب سازی اول بر اساس سن و بعد وزن و بعد نام ذخیره کند.
- تابع search : در لینکدلیست بر اساس نام وارد شده توسط کاربر جستجو کند و شماره(index) اولین رخدهی آن نام را چاپ کند.

به صورت کلی برنامه شما باید به این گونه باشد که در هر مرحله از کاربر سوال کند چه کاری می‌خواهد انجام دهد. سپس با توجه به کار درخواست شده از کاربر سوال های لازمه را بپرسد و تابع مورد نظر را انجام دهد. در ضمن این تمرین به صورت دستی تصحیح خواهد شد.