

به نام او



دوره ی پایانی المپیاد کامپیوتر ایران
فرودین ۱۳۸۷
آزمون تمرینی

Monsters Inc.

شرکت هیولاها



زمان مجاز برای هر مورد: ۴۰۰ میلی ثانیه^۱
حافظه ی مصرفی مجاز: ۴۴ مگابایت^۲

در شرکت هیولاها، n هیولا مشغول به کار هستند. کمیته بازیابی اطلاعات قصد دارد هیولاها را بر حسب میزان حقوقی که می گیرند مرتب کند. مشکل نحوه ی صحبت هیولاها با آدم ها است. ما نمی توانیم مستقیماً از هیولاها مقدار حقوقشان را بپرسیم. تنها سؤالی که می توانیم بپرسیم این است:

«آیا هیولای i کمتر از هیولای j حقوق می گیرد؟»

می دانیم هیچ دو هیولایی مقدار حقوق برابر ندارند. به کمیته بازیابی اطلاعات کمک کنید تا با تعداد کمی پرسش، هیولاها را بر حسب حقوق مرتب کند. شما می توانید حداکثر ۲۰۰۰۰۰۰ سؤال از هیولاها بپرسید و گرنه هیولاها عصبانی می شوند و همه چیز را نابود می کنند.

مسئله

برنامه ای بنویسید که

- تعداد هیولاها (n) را از کتابخانه ی پیوندی بخواند؛
- به کمک کتابخانه ی پیوندی سؤالات مورد نظر را از شرکت هیولاها بپرسد؛
- به کتابخانه ی پیوندی گزارش دهد ترتیب هیولاها بر حسب حقوق چگونه است.

کتابخانه

توابع و تعاریفی که توسط کتابخانه در اختیار شما قرار گرفته، به صورت زیر است:

```
int init();
```

^۱ زمان مجاز در واقع ۵۰۰ میلی ثانیه می باشد و کتابخانه حداکثر ۱۰۰ میلی ثانیه از آن را استفاده می کند.
^۲ حافظه ی مصرفی مجاز در واقع ۶۴ مگابایت می باشد و کتابخانه حداکثر ۲۰ مگابایت از آن را استفاده می کند.

این تابع n (تعداد هیولاها) را باز می‌گرداند ($1 \leq n \leq 10000$) و باید دقیقاً یک بار و پیش از دیگر توابع کتابخانه فراخوانی شود.

```
int compare(int i, int j);
```

این تابع یک سؤال (مقایسه) از هیولاها می‌پرسد و پاسخ‌شان را باز می‌گرداند. دو پارامتر ورودی این تابع، شماره‌ی دو هیولا^۳ است. خروجی این تابع برابر ۱ است اگر حقوق هیولای i کمتر از حقوق هیولای j باشد. در غیر این صورت، خروجی برابر ۰ است.

```
void report(int *index);
```

با فراخوانی این تابع می‌توانید پاسخ خود را به کتابخانه تحویل دهید. این تابع باز نمی‌گردد و پس از فراخوانی‌اش برنامه‌ی تان پایان می‌یابد. ورودی این تابع، یک اشاره‌گر است به ابتدای آرایه‌ای به طول n از نوع `int`. شماره‌ی هیولاها را به صورت مرتب شده برحسب حقوق در این آرایه قرار دهید^۴. این آرایه باید شامل جایگشتی از اعداد ۱ تا n باشد.

نقض شرایط گفته‌شده به منزله‌ی نقض استفاده‌ی درست از کتابخانه است. برنامه‌ی تان نباید با هیچ فایل‌ی کار کرده، از ورودی استاندارد بخواند یا به ورودی استاندارد بنویسد. در ضمن برنامه‌ی شما نباید سعی کند به فضایی خارج از محدوده‌ی خود^۵ دسترسی داشته باشد. بدیهی است که نقض هریک از این حدود می‌تواند منجر به حذف شدن از گردونه‌ی رقابت‌ها گردد.

دو فایل `sort.h` و `sortlib.cpp` به شما داده شده است. بالای برنامه‌ی تان باید عبارت `#include "sort.h"` را قرار دهید. برای هم‌گردانی^۶ برنامه‌ی تان فایل‌های فوق‌الذکر را در کنار برنامه‌ی خودتان (در این جا با نام `sort.cpp`) کپی کرده و از دستور

```
g++ -O2 -static sort.cpp sortlib.cpp -lm
```

استفاده کنید. فایل `ssort.cpp` نیز یک نمونه از کار کردن با این کتابخانه است. توجه به دو نکته ضروری است:

(۱) برنامه‌ی نمونه‌ی داده شده، یعنی `ssort.cpp` کار مفیدی انجام نمی‌دهد و تنها برای نمایش نحوه‌ی کار کردن با کتابخانه فراهم شده است.

(۲) در هنگام ارزیابی برنامه‌ها از کتابخانه‌ی دیگری استفاده خواهد شد که توابع و تعاریف درون `sort.h` را حمایت می‌کند. مادامی که مطابق دستورالعمل‌ها از کتابخانه استفاده می‌کنید، این موضوع بر برنامه‌ی شما تأثیری نخواهد داشت.

آزمایش

کتابخانه‌ی آزمایش را می‌توانید از واسط مسابقه دریافت^۷ کنید. کتابخانه‌ی آزمایشی داده شده، داده‌های مورد نیاز را از ورودی استاندارد می‌خواند. باید در سطر اول ورودی، n (تعداد هیولاها)، و در سطر بعد، n عدد متفاوت نوشته شده باشند که با فاصله از هم جدا شده‌اند. عدد i ام سطر دوم، مقدار حقوق هیولای شماره‌ی i را نشان می‌دهد.

هنگام استفاده از واسط آزمایش^۸ برنامه‌ی تان در برابر همین کتابخانه کار می‌کند و فایل‌ی که به عنوان

^۳شماره‌ها از ۱ تا n هستند.

^۴شماره‌ی هیولا با کم‌ترین حقوق در `index[0]` و هیولا با بیشترین حقوق در `index[n-1]` قرار می‌گیرد.

^۵به خصوص فضای مربوط به کتابخانه

^۶compile

^۷download

^۸test run

ورودی ارسال می‌کنید، باید دارای ساختار مذکور باشد.
شما آزادی برنامه‌هایی که در اختیارتان قرار داده شده را تغییر دهید ولی توصیه می‌شود که فایل sort.h را تغییر ندهید.

شیوه‌ی ارزیابی

اگر برنامه‌ی‌تان از محدودیت‌های اعمال شده تجاوز کند، یا از کتاب‌خانه به صورت شایسته استفاده ننماید، به‌ازای آن مورد آزمون، نمره‌ای به برنامه تعلق نمی‌گیرد.
برنامه‌تان حداکثر اجازه‌ی انجام ۲۰۰۰۰۰۰ پرسش را دارد. اگر بیش از این تعداد پرسش انجام دهید، برنامه‌تان متوقف شده و نمره‌ای نمی‌گیرید. اگر در پایان کار، گزارش داده‌شده (ترتیب هیولاها) نادرست باشد نیز نمره‌ای نمی‌گیرید. همچنین، اگر ترتیب هیولاها را به درستی گزارش دهید ولی با توجه به پرسش‌هایی که کرده‌اید تعیین ترتیب هیولاها به طور یکتا غیرممکن باشد، نمره‌ای برای آن مورد آزمون به برنامه داده نمی‌شود. برای هر مورد آزمون، در صورت دادن پاسخ نهایی درست، درصد نمره‌ای که از آن مورد آزمون دریافت می‌کنید، به اندازه‌ی

$$\min \left\{ 1, \frac{4n \lceil 1 + \log_2(n) \rceil}{questions + 1} \right\} \times 100\%$$

خواهد بود که در آن، $questions$ تعداد پرسش‌های برنامه‌ی شما از کتاب‌خانه است.

تعامل نمونه

در زیر یک تعامل ساده با کتاب‌خانه آمده است.

توضیحات	فراخوانی
مقدار n برابر خروجی تابع یعنی 3 می‌شود.	<code>int n=init();</code>
مقدار x برابر خروجی تابع یعنی 1 می‌شود.	<code>int x=compare(1, 2);</code>
مقدار y برابر خروجی تابع یعنی 1 می‌شود.	<code>int y=compare(1, 3);</code>
مقدار z برابر خروجی تابع یعنی 0 می‌شود.	<code>int z=compare(2, 3);</code>
تعریف یک آرایه از نوع <code>int</code> جهت دادن جواب نهایی.	<code>int a[3];</code>
مقدار دهی a .	<code>a[0]=1; a[1]=3; a[2]=2;</code>
جواب نهایی را اعلام می‌کنیم.	<code>report(a);</code>

یک نمونه از فایل ورودی برای `sortlib.cpp` را در زیر مشاهده می‌کنید.

```
10
1 3 7 2 4 9 6 5 8 0
```