

درج عقبکی

- محدودیت زمان: 1 ثانیه
- محدودیت حافظه: 100 مگابایت

در این سوال باید سورت درجی (insertion sort) را به گونه بازگشتی پیاده‌سازی کنید. بدین صورت که تابعی را پیاده‌سازی کنید که هر در مرحله با فراخوانی آن به صورت بازگشتی آرایه ای را به عنوان ورودی به آن تابع بدهید و در نهایت آرایه سورت شده را به عنوان خروجی دریافت کنید.

ورودی:

به عنوان ورودی ابتدا n را دریافت می کنید :

$$1 < n < 10^9$$

و سپس آرایه‌ای n تایی از اعداد صحیح به عنوان ورودی دریافت می کنید

خروجی:

آرایه را به صورت مرتب شده به عنوان خروجی چاپ می کند

ورودی نمونه :

5
6 8 2 1 2

خروجی نمونه:

1 2 2 6 8

نکته : کد های سابمیت شده برای این سوال به صورت دستی نیز بررسی خواهند شد و در صورتی که سوال به صورت گفته شده پیاده سازی نشده باشد نمره ای به کد ارسال شده تعلق نمی‌گیرد.

کابوس مرج سورت

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: 128 مگابایت

در این سوال از شما می خواهیم تا برنامه ای بنویسید تا آرایه از اعداد صحیح را دریافت کرده و آن آرایه را تبدیل به کابوسی برای مرج سورت کند (آرایه را به بدترین حالت ممکن (worst case) برای مرتب سازی ادغامی (مرج سورت خودمون) تبدیل کند) به صورتی که مرتب سازی آرایه خروجی بیشترین زمان را از الگوریتم مرتب سازی درجی بگیرد.

ورودی:

به عنوان ورودی ابتدا عدد n را دریافت می کنید :

$$1 < n < 10^9$$

و سپس در خط بعد آرایه ای n تایی از اعداد صحیح به عنوان ورودی دریافت می کنید

خروجی:

آرایه را به کابوس مرج سورت تبدیل کنید

مثال

ورودی نمونه

5
1 5 2 8 9

خروجی نمونه

1 9 5 2 8

سورت آگاهانه (امتیازی)

- محدودیت زمان: 0.4 ثانیه (بسیار کم)
- محدودیت حافظه: ۲۵۶ مگابایت (به میزان کافی)
- نوع : امتیازی

ورودی

در اولین خط ورودی به شما عدد n داده می‌شود که:

$$0 < n < 10^6$$

سپس در خط بعدی آرایه ای n عضوی از اعداد صحیح به شما داده می‌شود.

خروجی

در خروجی شما باید آرایه داده شده را به صورت سورت شده از کوچک به بزرگ چاپ کنید.

نکته‌ی مهم

الگوریتم مورد استفاده برای حل این سؤال باید ار در کوچک‌تری نسبت به $n \log(n)$ داشته باشد؛ دلیلش هم خیلی ساده اس وقت نداریم و در غیر اینصورت نمره‌ای برای حل در نظر گرفته نمی‌شود (:

مثال

ورودی نمونه ۱

1
10

خروجی نمونه ۱

10

ورودی نمونه ۲

10
9 8 7 6 5 4 3 2 1 0

خروجی نمونه ۲

0 1 2 3 4 5 6 7 8 9

معجزه ای به نام اردر (امتیازی)

تا این سوال شما دو الگوریتم مرتب سازی درجی و مرتب سازی ادغامی را پیاده سازی کرده اید (اگر هنوز کد این دو الگوریتم را پیاده سازی نکرده اید حتما این کار را انجام دهید و تابع هر دو نوع مرتب سازی در کد ارسالی برای این سوال موجود باشد) حال می‌خواهیم تاثیر $\log(n)$ را به صورت عملی ببینیم ابتدا آرایه 300000 تایی (این عدد می‌تواند با توجه به زبان برنامه نویسی مورد استفاده شما و یا قدرت پردازشی سیستم‌تان متفاوت باشد اما نتیجه می‌بایست مشهود باشد) از اعداد رندوم بین 10000- تا 10000+ ایجاد کنید. حال تایمری را ران کرده و این آرایه یکبار با الگوریتم مرج سورت $(n \log(n))$ و بار دیگر با ریست کردن تایمر با الگوریتم اینزرتشن سورت (n^2) اجرا کنید و در هر مرحله استفاده از هر یک از دو الگوریتم زمان اجرا را چاپ کنید.

می‌توانید این کار را برای آرایه ای با اعضای بیشتر انجام دهید تا با معجزه اردر ها بیشتر آشنا شوید.

فرمت خروجی ارسالی :

برای این تمرین می‌بایست یک فایل شامل یک کد و یک عکس را ارسال کنید

فایل کد :

یک فایل کد به دلخواه (cpp، پایتون، جاوا، #C) شامل دو تابع برای مرتب سازی درجی و مرتب سازی ادغامی به همراه موارد مختلف مربوط به ایجاد آرایه رندوم و تایمر گرفتن.

فایل عکس :

فایل عکس می‌بایست از خروجی کد شما و با این فرمت باشد :

```
insertion sort timer:5..... s
merge sort timer: 1..... s
```

```
insertion sort timer: 9.21 s
merge sort timer: 3.09 s
```