

مرتب سازی بیتی

در این مسئله، انتظار می‌رود یک آرایه از اعداد صحیح را بر اساس تعداد بیت‌های روشنشان به صورت صعودی مرتب کنید.

بیت ۱ بیت روشن نام دارد:))

ورودی

در تنها سطر ورودی n عدد دریافت می‌شود که با یک فاصله از یکدیگر جدا شده‌اند.

خروجی

در تنها سطر خروجی باید لیست مرتب شده را نمایش دهید.

مثال ۱:

ورودی

7 6 15 8

خروجی

[8, 6, 7, 15]

توضیحات

- 7 سه بیت روشن دارد (0111)
- 6 دو بیت روشن دارد (0110)
- 15 چهار بیت روشن دارد (1111)

• 8 یک بیت روشن دارد (1000)

بنابراین لیست به ترتیب [8, 6, 7, 15] مرتب می‌شود.

نکته: در مواردی که دو عدد تعداد بیت یکسانی دارند، مقادیر واقعی آن‌ها را با یکدیگر مقایسه کنید. به عنوان مثال، بین 10 (1010) و 12 (1100)، هر دو دارای تعداد بیت یکسانی، یعنی 2 است، اما عدد 10 کمتر از 12 است، بنابراین در لیست قبل از عدد 12 قرار می‌گیرد.

مثال ۲:

ورودی

3 8 3 6 5 7 9 1

خروجی

[1, 8, 3, 3, 5, 6, 9, 7]

روابط امروزی

از آنجایی که کاربرد گراف ها برهیچکس پوشیده نیست از شما می‌خواهیم در این تمرین گرافی جهت‌دار طراحی کنید تا در خود، نام افراد و روابط بین آنها را ذخیره کند. پس از ایجاد چنین گرافی باید بتوانید تشخیص دهید که بین فرد x و y رابطه‌ای وجود دارد یا نه.

ورودی:

در سطر اول تعداد سطرهای ورودی مشخص می‌شود و در n سطر بعدی ورودی، نام فرد و افرادی که با آنها در ارتباط است دریافت می‌شود. در سطر آخر دو اسم به شما داده می‌شود. اسم اول مبدا و اسم دوم مقصد جست‌جو می‌باشد.

توجه: نیازی به اعتبار سنجی داده ها نیست و صحت آنها تضمین می‌شود.

مثال ورودی

Ali#Mohammad-Zahra-Sara-Mohsen

در ورودی بالا گره Ali به گره های Mohammad , Zahra , Sara , Mohsen رابطه دارد.

خروجی:

در صورت وجود مسیر بین دو گره در تنها سطر خروجی مقدار True یا False چاپ کنید.

مثال ۱:

ورودی نمونه:

6

Ali#Mohammad-Zahra-Sara-Mohsen

Mohammad#Reza

Jafar#Sara
Mohsen#Ahmad
Ahmad#Arman
Arman#Reza
Ali-Reza

در ورودی نمونه بالا در سطر اول عدد ۶ وارد شده که نشان می‌دهد تعداد سطرهای ورودی گراف چند است!
در خط پایان نیست از ما می‌خواهد بررسی کنیم که آیا علی با رضا ارتباط دارد یا خیر!

خروجی نمونه:

True

توضیحات:

از آنجایی که گره Ali به Mohammad رابطه دارد و گره Mohammad نیز به گره Reza پس می‌توان نتیجه گرفت که گره Ali و Reza بایکدیگر در ارتباطند.

پردازش لاگ

از شما به عنوان برنامه‌نویس که به تازگی با ریجکس آشنا شده‌اید خواسته شده که با استفاده از ریجکس، تاریخ، ساعت و تعداد لاگ‌ها را پردازش کنید و یک الگوی مناسب در قالب یک تابع بنویسد.

ورودی

ورودی شامل دو مقدار است:

متغیر n که تعداد خطوط لاگ را نشان می‌دهد.

$$1 \leq n \leq 10^6$$

لاگ‌هایی که قرار است پردازش شوند.

خروجی

در این بخش لاگ پردازش شده با فرمت صحیح نشان داده شود.

Year {year}, Hour {hour}: INFO={counts['INFO']}, WARNING={counts['WARNING']},

مثال

ورودی نمونه ۱

3

[2020-07-28 08:04:19] ERROR: Alice write wrong regex.

[2020-07-28 08:04:19] ERROR: Alice make mistake.

[2024-01-04 23:17:55] INFO: Alice confused.

خروجی نمونه ۱

Year 2020, Hour 08: INFO=0, WARNING=0, ERROR=2

Year 2024, Hour 23: INFO=1, WARNING=0, ERROR=0

ورودی نمونه ۲

2

[2024-01-04 23:17:55] INFO: Alice confused.

[2023-09-18 12:03:18] WARNING: Alice does not use regex.

خروجی نمونه ۲

Year 2024, Hour 23: INFO=1, WARNING=0, ERROR=0

Year 2023, Hour 12: INFO=0, WARNING=1, ERROR=0

نکته

کد شما حتما باید در قالب یک تابع باشد.
برای حل سوال فقط باید از ریجکس استفاده شود.

مدیریت خطا


این سوال از دو قسمت تشکیل شده که برای هر قسمت باید توابع مخصوص به خود را کامل کنید .

قسمت اول:

یک فایل متنی در سیستم خود ایجاد کنید که در آن متن زیر قرار داشته باشد:

Hello, Quera!


سپس در تابع `read_file` مسیر فایل به عنوان ورودی بگیرید و آن فایل را بخوانید. شما باید تابع زیر را به گونه‌ای کامل کنید که خطاهای ممکن در زمان خواندن فایل را پوشش دهد. در زمان مناسب استثنای مربوط که در فایل `exception.py` وجود دارد را پرتاب کنید.

 source.py

```
1 | def read_file(file_path):  
2 |     pass
```

قسمت دوم:

در تابع `divide_numbers` دو عدد به عنوان ورودی می‌گیرد و آن‌ها را تقسیم می‌کند باید تمام خطاهای ممکن در زمان تقسیم دو عدد را پوشش دهد. در زمان مناسب استثنای مربوط که در فایل `exception.py` وجود دارد را پرتاب کنید.

 source.py

```
1 | def divide_numbers(a, b):  
2 |     pass
```

نکات

- می‌توانید فایل اولیه این سوال را از [این لینک](#) دریافت کنید.
- در توابعی که از شما خواسته شده مقداری را بازگردانید، باید مقدار مورد نظر را `return` کنید.
- توجه داشته باشید که این سوال پس از اتمام مدت زمان تمرین به صورت دستی نمره داده خواهد شد.

نحوه ارسال

شما باید توابع موجود در فایل `source.py` را تکمیل و سپس ارسال کنید.

مرتب سازی بازگشتی

در این مسئله، انتظار می‌رود یک آرایه از اعداد صحیح را با استفاده از تابع بازگشتی و قاعده *merge sort* به صورت صعودی مرتب کنید.

ورودی

در تنها سطر ورودی n عدد دریافت می‌شود که با یک فاصله از یکدیگر جدا شده‌اند.

خروجی

در هر سطر قسمت سمت چپ را قبل از مرتب سازی پرینت کنید و در سطر انتهایی لیست مرتب شده نهایی را چاپ کنید

مثال ۱:

ورودی

12 11 13 5 6 7

خروجی

[11]

[12]

[6]

[5]

[11, 12, 13]

[5, 6, 7, 11, 12, 13]

مثال ۲:

ورودی

5 3 8 4 2 7 1 9

خروجی

[5]

[8]

[3, 5]

[2]

[1]

[2, 7]

[3, 4, 5, 8]

[1, 2, 3, 4, 5, 7, 8, 9]

توضیحات

در این مسئله نحوه کدنویسی شما مورد بررسی قرار می‌گیرد و شما باید در کد خود از تابع بازگشتی برای مرتب کردن آرایه با استفاده از قاعده تقسیم و ادغام که یکی از سریع ترین روش های مرتب سازی آرایه می‌باشد استفاده بفرمایید. در صورتی که از این قواعد استفاده نکرده باشید از نمره شما کسر می‌شود. (تا ۵۰ امتیاز امکان کسر شدن وجود دارد)