به نام خدا

هانیه اسعدی ۹۹۵۲۱۰۵۵

منابع در فایل .ipynb‌ ذکر شده است.

۱- از تابع Where برای حل سوال استفاده می‌کنیم. هر عنصر را با عنصر متناظر مقایسه کرده و با توجه به شرط، مقدار true/false را در آرایه جواب قرار می‌دهیم. برای مثال برای آرایه بزرگتر، در مقایسه عنصر به عنصر، اگر عنصر در آرایه 1 از عنصر متناظر در آرایه 2 بزرگتر باشد، مقدار true در آرایه جواب قرار می‌دهیم.

2- براساس متد گرفته شده در ورودی تابع، اگر متد element-wise باشد، از تابع multiply که برای ضرب عنصر به عنصر است استفاده می‌کنیم. در غیر این صورت، از تابع matmul که برای ضرب ماتریسی دو ماتریس است استفاده می‌کنیم.

3- همانند سوال قبل، بر اساس متد گرفته شده عمل می‌کنیم. اگر متد، row-wise بود، ماتریس دوم ورودی را به صورت سطر به ماتریس اول اضافه می‌کنیم. برای این کار از تابع vstack استفاده می‌کنیم.ورودی تابع یک تاپل است که عضو اول آن، ماتریسی است که میخواهیم به آن ردیف اضافه کنیم و عضو دوم تاپل، ماتریسی است که آن را قرار است اضافه کنیم. و اگر متد column-wise بود، ماتریس دوم را به صورت عمودی در ستون آخر ماتریس اول اضافه می‌کنیم. برای اضافه کردن عمودی ماتریس، ابتدا باید ماتریس افقی داده شده را به عمودی تبدیل کنیم. برای اینکار از تابع reshape استفاده می‌کنیم. ماتریس تبدیل شده باید n سطر و یک ستون داشته باشد. پس از تبدیل ماتریس دوم، برای اضافه کردن ستون به ماتریس اول، از تابع hstack استفاده می‌کنیم که مشابه تابع vstack ورودی دریافت می‌کند. اگر متد ورودی چیزی جز این دو مورد باشد

، invalid input method در خروجی چاپ می‌شود.

4- با استفاده از random.randint یک ماتریس 4 در 4 با عناصر بین 1 تا 10 تولید می‌کنیم. برای نرمال کردن مقادیر، ابتدا مقادیر ماکسیمم و مینیمم ماتریس را پیدا می‌کنیم. برای نرمال کردن هر عنصر، اختلاف عنصر و مقدار مینیمم را بر اختلاف مقدار ماکسیمم و مینیمم تقسیم می‌کنیم.

7- مثل سوال اول، هر عنصر را با threshold داده شده مقایسه کرده، اگر مقدار کوچکتر یا مساوی ترشلد باشد، مقدار صفر و در غیر این صورت، 1 جایگزین میشود.