

زمان بندی

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

در یک سیستم کامپیوتری، مجموعه‌ای از n پردازش داریم که هر پردازش زمان ورود مشخصی داشته و مدت زمان معینی هم طول می‌کشد اما بعد از یک زمان مشخص T از ورود پردازش، نتیجه پردازش‌ها به درد نمی‌خورند. زمان ورود و طول زمان پردازش i را به ترتیب با S_i و L_i نمایش می‌دهیم (این اعداد از نوع صحیح مثبت هستند). هنگامی که یک پردازش در حال اجراست، سیستم عامل می‌تواند پردازش را متوقف کرده و پردازش دیگری را اجرا نماید، در این صورت پردازش اول برای ادامه اجرا باید مجدداً زمان‌بندی شود (مقداری از پردازش که تاکنون اجرا شده ذخیره می‌شود). در ضمن، در هر لحظه نیز فقط یک پردازش را می‌توان اجرا نمود. بنابراین، در صورتی که پردازشی وارد شود، باید احتمالاً مدت زمانی را در حالت انتظار باشد تا نوبت به وی برسد.

با فرض اینکه هنگام ورود یک پردازش مدت زمان اجرای آن مشخص است، سیستم عامل باید ترتیب اجرای پردازش‌ها را مشخص نماید به نحوی که بیشترین تعداد پردازش‌ها انجام شوند (توجه کنید که سیستم عامل از ابتدا نمی‌داند که چه زمانی پردازش‌ها ورود پیدا می‌کنند، بلکه با ورود هر پردازش متوجه ورود و طول زمان پردازش آن می‌شود). برای مثال، فرض کنید پردازش‌های زیر داده شده باشند:

$$T=3$$

$$S_1 = 0, L_1 = 6$$

$$S_2 = 1, L_2 = 3$$

$$S_3 = 2, L_3 = 2$$

$$S_4 = 3, L_4 = 1$$

در این صورت، سیستم عامل پردازش‌ها را به ترتیب ۱، ۲، ۴ اجرا می‌نماید که در این حالت دو پردازش به نتیجه مورد نظر ما رسیده‌اند. الگوریتمی با کمترین زمان اجرا و به صورت حریصانه و بازگشتی ارائه دهید که بتواند مسئله فوق را حل کند.

ورودی

ورودی برنامه شامل تعدادی خط است که در خط اول مقدار T از ورودی خوانده می‌شود و سپس در هر خط دو عدد صحیح داده شده است، که به ترتیب زمان ورود و طول اجرا را نشان می‌دهند. زمان ورود پردازش‌ها منطقی است و زمان اجرای هیچ پردازشی صفر نیست.

خروجی

صرفاً باید تعداد پردازش‌های موفق را در خروجی چاپ کنید.

مثال

ورودی نمونه ۱

3
0 6
1 3
2 2
3 1

خروجی نمونه ۱

2

در مثال بالا، ابتدا پردازش شماره یک به مدت یک ثانیه اجرا شده است. سپس متوقف شده و پردازش شماره دو به مدت ۳ ثانیه اجرا شده (در این مدت پردازش‌های ۳ و ۴ وارد می‌شوند و به همراه ۱ منتظر

می‌مانند اما پس از دو ثانیه از اجرای پردازش 2، نتیجه پردازش یک به درد نمی‌خورد و حذف می‌شود). سپس پردازش 4 به مدت ۱ ثانیه اجرا می‌شود (در انتهای تکمیل پردازش 4، نتیجه پردازش 3 دیگر به درد ما نمی‌خورد). بنابراین تعداد پردازشهای موفق، 2 است.