

## ربات سرگردان

سال ۲۲۳۵ است. بشریت موفق شده ربات‌های مینیاتوری بسازد که می‌توانند منابع انرژی کوانتومی پراکنده در فضا را جمع‌آوری کنند. این منابع انرژی در مسیرهای ناشناخته‌ای قرار دارند و هر عدد انرژی یک ویژگی خاص دارد.

ربات‌ها یک قانون عجیب دارند:

هر منبع انرژی تنها در صورتی قابل جمع‌آوری است که با دو منبع قبلی رابطه خاصی داشته باشد.

عدد اول و دوم مسیر همیشه جمع‌آوری می‌شوند، اما از عدد سوم به بعد، قوانین سخت‌گیرانه‌تری اعمال می‌شود.

تو به عنوان برنامه‌نویس مامور شدی ربات‌ها را هدایت کنی تا انرژی را جمع کنند.

### قوانین جمع‌آوری انرژی:

1. یک تابع به نام `canCollect(prev1, prev2, current)` باید بررسی کند که آیا عدد فعلی (`current`) قابل جمع‌آوری است یا نه.

2. قوانین تابع:

#### قانون اول :

جمع ارقام `current` باید بزرگ‌تر یا مساوی جمع ارقام هر یک از دو عدد قبلی باشد.

#### قانون دوم :

عدد `current` باید یا مضربی از بزرگ‌ترین رقم موجود در `prev1` و `prev2` باشد، یا مجموع ارقامش مضربی از ۵ باشد.

#### وظیفه شما:

برنامه‌ای بنویسید که:

1.تعداد نقاط انرژی را از ورودی بگیرد

2.مقادیر انرژی هر نقطه را بخواند

3.با استفاده از قوانین تابع canCollect، مجموع انرژی قابل جمع‌آوری را محاسبه کند

### ورودی:

یک عدد صحیح N (تعداد نقاط انرژی)

سپس N عدد صحیح مثبت، هر عدد ارزش انرژی در آن نقطه

### خروجی:

یک عدد صحیح: مجموع انرژی‌هایی که ربات می‌تواند جمع‌آوری کند

### نمونه ورودی :

10

33 21 12 19 28 28 10 14 49 37

### خروجی :

129

توضیح:

عدد اول و دوم جمع می‌شوند : جمع = 54

عدد سوم نداشتن قانون اول

عدد چهارم هر دو قانون را دارد : جمع = 73

عدد پنجم هر دو قانون را دارد : جمع = 101

عدد ششم هر دو قانون را دارد : جمع 129

عدد هفتم نداشتن قانون اول و دوم

عدد هشتم نداشتن قانون اول

عدد نهم نداشتن قانون دوم

عدد دهم نداشتن قانون اول

نکته قابل توجه این است که در شرط دوم هنگام حساب کردن بزرگ ترین رقم باید هر دو عدد رو در نظر بگیرید

و بزرگ ترین رقم رو در نظر بگیرید نه جدا جدا رقم بزرگشون رو حساب کنید !!

بزرگ ترین رقم 37 و  $43 = 7$