



## مسئله‌ی ۱. سرورهای edu

گراف داده شده درخت است. اگر  $m$  را برابر با بزرگترین مقدار  $a_i$  ها در نظر بگیریم جواب نهایی تنها میتواند برابر با  $m$  یا  $m + 1$  و یا  $m + 2$  باشد. حال فرض کنید که  $u$  اولی سروری است که قصد خراب کردنش را داشته باشیم. ما نیاز به قدرت حداقل  $a_u$  برای این کار داریم. حال این مقدار برای همسایه های  $u$  مانند  $v$  برابر است با  $a_v + 1$  و برای سایر سرورها که هنوز خراب نشده اند مانند  $x$  برابر است با  $a_x + 2$ . حال با پیمایش و ذخیره ی مقدار ماکسیموم میتوان جواب را به دست آورد.

## مسئله‌ی ۲. پرررو!

برای حل این سوال از پیمایش inorder استفاده میکنیم. حال سوال تبدیل می شود به یافتن کمترین جا به جایی ها برای تبدیل یک آرایه به آرایه ای مرتب شده. به این منظور میتوان سوال را به یک گراف تناظر داد که از راس  $i$  به راس  $j$  یال وجود دارد اگر در آرایه ی مرتب شده عدد موجود در ایندکس  $i$  در آرایه ی ابتدایی در ایندکس  $j$  ام ظاهر شود. جواب نهایی برابر است با  $\sum (cyclesize - 1)$

## مسئله‌ی ۳. دیکشنری خوب!

برای حل این سوال از داده ساختار ترای استفاده می کنیم. به این صورت که برای هر نود عددی نگه داشته می شود که نشان دهنده ی بیشترین تعداد تکرار یک کلمه در زیر درخت مربوط است.

## مسئله‌ی ۴. ارتش آزادی بخش pubg

برای حل این سوال از باینری ترای استفاده میکنیم. به این منظور، تمامی  $Pi$  ها را به صورت فرم باینری در ترای ذخیره میکنیم. اکنون برای هر زیردرخت مقداری که نمایانگر تعداد  $Pi$  هایی که در آن زیردرخت موجود هستند را به دست می آوریم. اکنون ترای را یک بار به سمت پایین پیمایش می کنیم و سعی می کنیم مقدار  $Pi \text{ XOR } Li$  را پیدا کنیم. زمانی که به یک زیردرخت می رویم مشخص می کنیم که آیا مقادیر زیردرختی که به آن وارد نمی شویم را وارد کرده ایم یا خیر. (با چک کردن این که بیت کنونی در  $Li$  یک هست یا نه) جواب سوال برابر است با مجموع سائز زیردرخت هایی که در هنگام پیمایش اضافه کردیم.