

بسمه تعالی

تمرین سری هفتم طراحی الگوریتم

الگوریتم های مریضانه

مساله بهینه سازی حافظه نهان:

در این مساله ما با یک حافظه نهان cache روبرو هستیم که fully-way-associative است.

ظرفیت این حافظه نهان برابر با k است. n درخواست خواندن از حافظه RAM داریم که ترتیب آنها در ورودی داده شده است.

هنگامی که داده درخواستی در cache وجود داشته باشد، cache hit اتفاق می افتد.

هنگامی که داده درخواستی در cache وجود نداشته باشد، cache miss اتفاق می افتد. در این صورت به صورت خودکار و اجباری داده درخواستی از حافظه اصلی RAM خوانده میشود و وارد cache می شود و در صورتی که cache پر باشد، الزاما یکی از داده های داخل cache خارج می شود.

(فرض کنید CPU فقط می تواند داده ها را از cache بخواند در نتیجه برای خواندن داده، می بایست داده داخل cache قرار داشته باشد، در نتیجه باید حتما داده درخواستی را به cache اضافه کنید.)

(با توجه به اینکه در این مساله تمام درخواست ها در آینده را میدانیم، ممکن است درخواست داده 17 ام فقط یکبار انجام شود، در این صورت آوردن این داده به cache در حالت عادی کاری مفید نخواهد بود. با این حال در این مساله شما ضرورتا باید در آن مرحله داده 17 ام را وارد cache خود بکنید. زیرا CPU شما در این مساله تنها از این طریق می تواند داده 17 ام را بخواند. و دسترسی مستقیم به حافظه اصلی ندارد.)

در این مساله از شما میخواهیم دو استراتژی حریصانه ارائه دهید به صورتی که:

هر کدام از برنامه های شما، با گرفتن اندازه k ، همچنین ترتیب درخواست ها، مشخص کند در هر کدام از n مرحله، cache چه وضعیتی دارد، همچنین در خط آخر مشخص کند که تعداد خواندن از حافظه اصلی RAM چقدر بوده است. و هدف برنامه شما این است که تعداد خواندن از حافظه اصلی RAM حداقل شود.

در انتها شما باید نتایج حاصل از این دو برنامه را نیز با هم مقایسه کنید.

توجه کنید در صورتی که یکی از برنامه های پیشنهادی شما، جواب بهینه را بدهد، نمره اضافی خواهد داشت.

در این صورت باید بتوانید اثبات کنید که الگوریتم شما الزاما جواب بهینه را میدهد.

خط اول ورودی عدد k و سپس n به ترتیب داده میشود.

در خط بعدی به ترتیب n درخواست داده میشود.

در خروجی نیز باید در n خط اول، وضعیت حافظه نهان در انتهای درخواست i ام را چاپ کنید.

در خط بعدی نیز تعداد خواندن از حافظه RAM را چاپ کنید.

ورودی نمونه:	2 7 a b c b c a a
خروجی نمونه:	a: a b: a b c: c b b: c b c: c b a: c a a: c a 4

ورودی نمونه:	2 10 a b c a a c b b b b
خروجی نمونه:	a: a b: a b c: a c a: a c a: a c c: a c b: b c b: b c b: b c b: b c 4

ورودی نمونه:	10 6 a b c d c e
خروجی نمونه:	a: a b: a b c: a b c d: a b c d c: a b c d e: a b c d e 5

در این مثال، به ازای تمام حرف های a، b، ...، e یک خواندن از حافظه اصلی داشتیم.