



تمرین کامپیوتری سوم



سیستم‌های عامل - پاییز ۱۳۹۹

گزارش کار

دانشکده مهندسی برق و کامپیوتر

نام و نام خانوادگی:

تاریخ:

استاد:

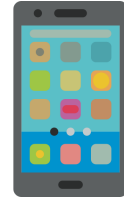
احسان قرائی

دکتر مهدی کارگهی

اشا سروش پور

2	مقدمه
3	پیاده‌سازی سری
3	سوال اول
3	سوال دوم
3	جدول اول
3	پیاده‌سازی چندریسه‌ای
3	سوال سوم
4	سوال چهارم
4	سوال پنجم
4	جدول دوم

مقدمه



در این تمرین به تحلیل داده‌هایی که از مشخصات و قیمت فروش گوشی‌های موبایل جمع‌آوری شده‌است پرداخته شده است. در ابتدا برنامه اقدام به خواندن و تجزیه مجموعه داده¹ی ارائه شده می‌کند و آنها را در حافظه خود ذخیره می‌کند. پس از استخراج داده‌ها و ویژگی‌های آنها، برنامه اقدام به نرمال‌سازی² داده‌ها و در نهایت اقدام به تعیین طبقه قیمتی گوشی‌ها می‌کند. این تمرین به دو روش این مسئله پیاده‌سازی شده است که در ادامه گزارش، نتایج حاصل آمده است.



¹ Dataset

² Data Normalization

پیاده‌سازی سری

سوال اول

چرا برای پیاده‌سازی یک برنامه بصورت چندریسه‌ای، بهتر است ابتدا این برنامه بصورت سری پیاده‌سازی شود؟
برای اینکه بفهمیم که چه بخشی از برنامه نیاز به این دارد که باید چندریسه‌ای باشد و اگر برنامه در زمان کوتاهی اجرا شود گزینه سری بودن گزینه بهتری می‌باشد چون در حالت چندریسه‌ای یک انتظار برای اتمام کار بقیه ریشه‌ها نیز زمان می‌برد.

سوال دوم

با بررسی زمان اجرای بخش‌های مختلف برنامه، Hotspot³ های برنامه را مشخص کنید.
در کل، حل پروژه داخل یک تابع قرار دارد که به صورت چندریسه‌ای (در پایین توضیحات بیشتری داده شده است) اجرا می‌شود اما اگر پروژه را به چند بخش تقسیم کنیم بخش ساختن داده Normalize شده طبق محاسبه پیش بینی گفته شده و پیدا کردن مقادیر min و max برای هر ستون Hotspot های برنامه می‌باشند.

جدول اول

زمان‌های اجرای ۶ اجرای متوالی از برنامه و میانگین آن‌ها را بازای ورودی نمونه‌ای که در شرح تمرین آمده است، در جدول زیر بیاورید.

میانگین	اجرای ششم	اجرای پنجم	اجرای چهارم	اجرای سوم	اجرای دوم	اجرای اول
۰.۰۳۴	۰.۰۳۴	۰.۰۳۲	۰.۰۳۳	۰.۰۳۶	۰.۰۳۵	۰.۰۳۴

³ توابی که در برنامه‌تان بیشترین زمان اجراها را به خود اختصاص می‌دهند.

پیاده‌سازی چندریسه‌ای

سوال سوم

اگر هنگام موازی‌سازی برنامه به زمان اجرای بیشتری نسبت به حالت سری برخورد کنید، چه رویکردهایی را برای کاهش زمان اجرا و استفاده حداکثری از موازی‌سازی پیش می‌گیرید؟

سعی میکنیم که محاسبه‌های تکراری را بیرون از ریشه‌ها انجام بدهیم و یا در صورتی که از متغیر مشترک زیاد استفاده میکنیم بهتر است که تغییرات را ذخیره کنیم و در نهایت به صورت یکجا تغییرات را اعمال کنیم تا بقیه Threadها معطل نشن و در نهایت حجم Threadها متعادل کنیم.

سوال چهارم

در هنگام پیاده‌سازی این بخش، به چه چالش‌هایی برخورد کردید و بیان کنید که به چه صورت آن‌ها را رفع کردید.

به غیر از چالش دیباگ کردن کد و پیاده‌سازی برنامه به صورت سری، چالش موازی‌سازی شامل این بود که در حین استفاده از مقادیر Global اگر دو thread جدا تصمیم به استفاده همزمان از یک متغیر مشترک گرفتند باعث می‌شد که برنامه Segmentation Fault می‌شد و یا در حین اینکه هربار میخواستیم از فایل داده شده token بسازیم همین مشکل باز پیش می‌آمد که به ترتیب با استفاده از Mutex و تابعی از C که توکن ساختن را با اشاره‌گر می‌ساخت، چالش‌های پیش آمده را رفع کردیم.

سوال پنجم

با توجه به تجربه‌ای که در پیاده‌سازی این تمرین بدست آوردید، به نظر شما در چه مواقعی از قفل⁴ در یک طراحی چندریسه‌ای ضروری است؟ تاثیر استفاده از قفل‌ها را بر روی کارایی⁵ سامانه بیان کنید.

وقتی که از مقادیر مشترک داخل برنامه در ریشه‌ها استفاده شود بهتر است که از قفل استفاده کنیم. از طرفی باید زمانی که یک ریشه منتظر می‌ماند کم باشد، چون در این صورت concurrency را از دست می‌دهیم و حتی از یک برنامه‌ای که به صورت سری پیاده‌سازی شده کندتر می‌شویم.

⁴ Lock

⁵ Performance

جدول دوم

زمان‌های اجرای ۶ اجرای متوالی از برنامه و میانگین آن‌ها را بازای ورودی نمونه‌ای که در شرح تمرین آمده است، در جدول زیر بیاورید.

میانگین	اجرای ششم	اجرای پنجم	اجرای چهارم	اجرای سوم	اجرای دوم	اجرای اول
۰.۰۱۸	۰.۰۱۷	۰.۰۱۸	۰.۰۱۹	۰.۰۱۹	۰.۰۱۸	۰.۰۱۷

میزان تسریع ($\frac{Serial\ Time}{Parallel\ Time}$) برنامه نسبت به حالت سری را در زیر بیاورید.

میزان تسریع	میانگین زمان اجرای موازی	میانگین زمان اجرای سری
۱.۸۸۸	۰.۰۱۸	۰.۰۳۴