



تمرین کامپیوتری سوم



سیستم‌های عامل - پاییز ۱۳۹۹

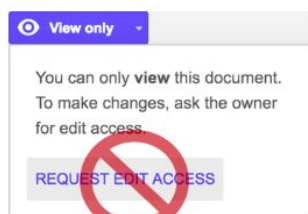
گزارش کار

دانشکده مهندسی برق و کامپیوتر

نام و نام خانوادگی:
سینا نگارنده

تاریخ:
۱۳۹۹/۰۹/۲۴

استاد:
دکتر مهدی کارگهی

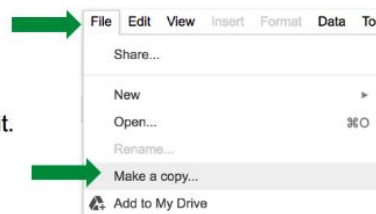


How to use this template:

This is a view-only file and cannot be edited.

Create your own copy of this template to edit.

In the menu, click **File > Make a copy...**



2

مقدمه

3

پیاده‌سازی سری

3

سوال اول

3

سوال دوم

3

جدول اول

3

پیاده‌سازی چندریسه‌ای

3

سوال سوم

4

سوال چهارم

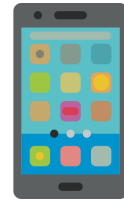
4

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 چندریشه‌ای، بهتر است ابتدا این برنامه بصورت سری پیاده‌سازی شود؟ پیدا کردن مشکلات برنامه (debug) کردن در حالت پیاده‌سازی موازی سخت است و با هر بار اجرا شاید خروجی متفاوتی بگیریم، به طور کلی نیز برنامه نویسی در حالت موازی سخت‌تر است، به همین دلیل بهتر است ابتدا برنامه به صورت سری پیاده‌سازی شود و بعد از اطمینان پیدا کردن از اجرای صحیح برنامه، برنامه به صورت موازی پیاده‌شود. همچنین با پیاده‌سازی به صورت سری تشخیص hotspot-های برنامه راحت‌تر است.

سوال دوم

با بررسی زمان اجرای بخش‌های مختلف برنامه، ³Hotspot های برنامه را مشخص کنید. قسمت‌هایی مانند خواندن فایل csv و ذخیره کردن اطلاعات آن در برنامه، پردازش کردن این اطلاعات مانند normalize کردن که شامل پیدا کردن بیشترین و کمترین مقدار برای هر ستون و همچنین انجام عملیات بر روی هر داده و یا انجام ضرب داخلی برای classify کردن و ... است.

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
0.735	0.74	0.73	0.73	0.74	0.76	0.71

³ توبعی 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؟

ممکن است چند دلیل داشته باشد، ممکن است اندازه dataset کوچک باشد و یا با ساخت و حذف مکرر قسمت‌های موازی باشد. علت دیگر می‌تواند استفاده از blocking call-ها باشد که با تبدیل کردن آن‌ها به non-blocking در صورت امکان می‌توانیم این مشکل را رفع کنیم. یک علت نیز می‌تواند این باشد که تعداد thread-های ساخته شده بیشتر از توان پردازشی موازی سیستم باشد که با کاهش تعداد thread-ها می‌توان این مشکل را نیز رفع کرد.

سوال چهارم

در هنگام پیاده‌سازی این بخش، به چه چالش‌هایی برخورد کردید و بیان کنید که به چه صورت آن‌ها را رفع کردید. چون برنامه به صورت موازی اجرا می‌شد و خروجی با هر بار اجرا متفاوت می‌شد تعیین دقیق مکان خطا کمی سخت بود. با قرار دادن break point، خروجی‌ها و مطالعه دقیق‌تر کد مکان خطا پیدا می‌شد. برای بعضی قسمت‌ها که (Mutual Exclusion) وجود داشت و همچنین طراحی حالتی که تعداد استفاده از lock-ها در کمترین مقدار باشد نیاز به دقت داشت. ابتدا از متغیرهای محلی استفاده و سپس برای هر thread تنها یک بار از lock استفاده کرده.

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.

زمانی که دو یا چند thread می‌خواهند به متغیری، داده و یا ... دسترسی پیدا کنند، تنها یکی از آن‌ها موفق می‌شود به آن دسترسی پیدا کند و بقیه thread-ها باید صبر کنند. در واقع نقطه‌ای برای همگام‌سازی است. اگر هر دو در یک زمان دسترسی پیدا کنند شاید یکدیگر را overwrite کنند یا به یکدیگر وابسته باشند و نتیجه اشتباه شود. به طور خیلی ساده می‌توان قفل را از دو قسمت تشکیل دهنده در نظر گرفت یکی flag و یک صف برای نگاه داشتن سایر thread-هایی که می‌خواهند دسترسی داشته باشند. در

⁴ Lock

⁵ Performance

صورتی که صف خالی باشد تنها flag تغییر می‌کند اما اگر صف خالی نباشد باید با استفاده از سیستم‌کال برای بیدار کردن نیاز داریم که باعث وقفه (stalls) می‌شوند، به همین دلیل قفل‌ها آهسته اجرا می‌شوند و استفاده زیاد از آن‌ها می‌تواند اجرای برنامه را کند کنند.

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
0.56	0.53	0.56	0.51	0.57	0.51	0.68

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
1.3125	0.56	0.735