



تمرین کامپیوتری سوم



سیستم‌های عامل - پاییز ۱۳۹۹

گزارش کار

دانشکده مهندسی برق و کامپیوتر

نام و نام خانوادگی:

حمیدرضا خدادادی

810197499

تاریخ:

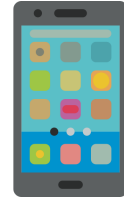
آذرماه 99

استاد:

دکتر مهدی کارگهی

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؟

جواب:

اگر برنامه را ابتدا به صورت سری پیاده‌سازی کنیم، می‌توانیم بهتر تشخیص بدهیم که کدام قسمت‌های برنامه قابل پیاده‌سازی با روش چند ریشه‌ای خواهد بود و به ما در پیاده‌سازی سریع‌تر چند ریشه‌ای کمک خواهد کرد.

سوال دوم

با بررسی زمان اجرای بخش‌های مختلف برنامه، ³Hotspot های برنامه را مشخص کنید.

جواب:

read_csv_file که دیتاها را از فایل‌ها می‌خواند.
normalization که ماکسیمم و مینیمم هر ستون را پیدا کرده و سپس دیتاهای خوانده شده از فایل train را نرمالایز می‌کند.
classification که مقدار score را برای هر سطر فایل train را به ازای 4 سطر موجود در فایل weights پیدا کرده و جواب را پیش‌بینی کرده و سپس با جواب هدف مقایسه می‌کند.

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.

زمان‌ها به ثانیه هستند.

میانگین	اجرای ششم	اجرای پنجم	اجرای چهارم	اجرای سوم	اجرای دوم	اجرای اول
0.047	0.058	0.050	0.034	0.034	0.065	0.041

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؟

جواب:

دوباره در تقسیم کارهای قابل موازی‌سازی بررسی و تجدید نظر می‌کنیم. در تعداد ریس‌ها تجدید نظر می‌کنیم چون ممکن زمان ساخت و کار کردن آن‌ها سرپار ایجاد کند. بخش‌هایی که به mutex lock لازم دارد را جوری پیاده‌سازی کرده که نیاز به توقف کار سایر ریس‌ها نشود.

سوال چهارم

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

جواب:

من در ابتدا از 5 تا ترد برای پیاده‌سازی این بخش استفاده کرده بودم که بنا به توضیحات سوال سوم لازم شد تعداد آن‌ها را به 4 تا کاهش بدهم.

در چند جا لازم بود همه‌ی ترد‌ها از یک داده استفاده کنند (مانند استفاده از آدرس پوشه‌ی دیتاها برای خواندن دیتاها، استفاده از ماکسیمم و مینیمم دیتاهای ستون‌ها برای نرمالایز کردن) در نتیجه این دیتاها را گلوبال تعریف کردیم.

مشکلی دیگر این که یک وکتور گلوبال برای نگهداری ماکسیمم و مینیمم دیتاهای هر ستون در دیتاهای مختص هر ترد داشتم که چون سایر تردها به نتیجه‌ی ترد دیگر وابسته بودند از mutex lock استفاده کردم و روش‌های دیگری نیز برای حل این مشکل وجود داشت.

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.

⁴ Lock

⁵ Performance

جواب:

استفاده از قفل در مواقعی که ترد ها از یک دیتای مشترک استفاده می کنند یا زمانی که سایر تردها از دیتای یک ترد دیگر استفاده می کند، ضروری است.

اما استفاده ی زیاد از قفل باعث کاهش کارایی برنامه می شود چون سایر ترد ها تا وقتی یک قفل توسط ترد مربوطه باز نشود متوقف خواهند شد.

جدول دوم

زمان های اجرای ۶ اجرای متوالی از برنامه و میانگین آن ها را بازای ورودی نمونه ای که در شرح تمرین آمده است، در جدول زیر بیاورید.

زمان ها به ثانیه هستند.

میانگین	اجرای ششم	اجرای پنجم	اجرای چهارم	اجرای سوم	اجرای دوم	اجرای اوّل
0.0285	0.028	0.031	0.029	0.027	0.029	0.027

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
1.649	0.0285	0.047