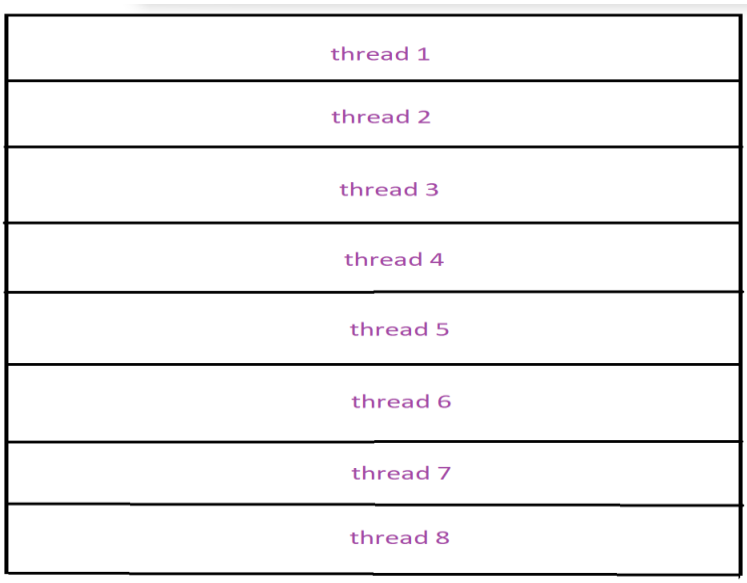
گزارش پروژه 3 سیستم عامل

آوا میرمحمدمهدی 810199501

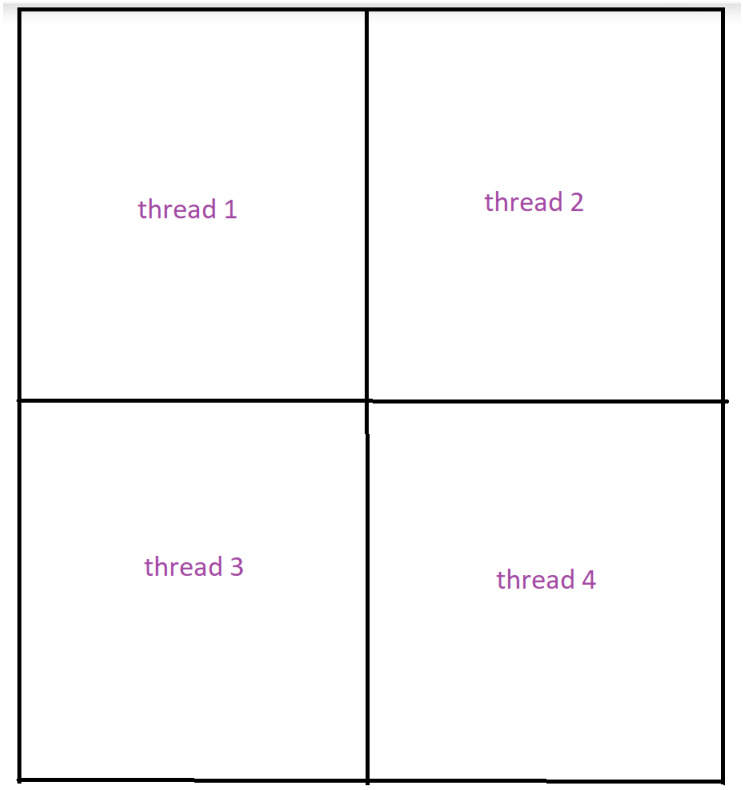
زمان اجرای بخشهای مختلف برنامه در حالت سری به ازای فایلی با اندازه 1500 در 3000 در زیر آورده شده است:

```
root@Ava:/mnt/d/Term5/OS/ca3/serial# ./ImageFilters.out os.bmp
getPixlesFromBMP24 Execution Time: 70 ms
mirrorFilter Execution Time: 21 ms
checkeredFilter Execution Time: 477 ms
diamondFilter Execution Time: 82 ms
writeOutBmp24 Execution Time: 87 ms
Execution Time: 776 ms
```

همانطور که می‌بینیم بخش شطرنجی کردن تصویر زمان زیادی را به خود اختصاص داده است؛ پس به موازی‌سازی برنامه با استفاده از ریسه‌ها می‌پردازیم. برای ذخیره فایل به فرمت RGB، ماتریسی که در آن RGB ذخیره می‌شوند را بر 8 تقسیم می‌کنیم به طوری که هر ریسه مسئول ذخیره‌سازی 1/8 از سطرهای فایل باشد، برای آیینه کردن تصویر و نوشتن در فایل نیز همین‌کار را تکرار می‌کنیم و عملیات روی هر قسمت از ماتریس توسط یک ریسه انجام می‌شود پس به طور کلی 8 ریسه داریم که عملیات یکسانی را انجام می‌دهند. تصویر فرضی و قسمت‌هایی که برای پردازش به هر ریسه می‌رسد در زیر نمایش داده شده است.



برای قسمت ترسیم لوزی، تصویر را به چهار قسمت شمال شرقی، شمال غربی، جنوب شرقی و جنوب غربی تقسیم می‌کنیم و با توجه به شماره آیدی هر ریسه بررسی یکی از چهار خطوط (که در قسمتی از تصویر که به آن ریسه داده شده است قرار دارد) را به عهده‌ی آن ریسه می‌گذاریم. در واقع خطی و قسمتی از تصویر که هر ترد به مسئولیت آن را دارد، متفاوت است.



زمان‌های اندازه‌گیری شده برای برنامه موازی به صورت زیر است:

```
root@Ava:/mnt/d/Term5/OS/ca3/parallel# ./ImageFilters.out os.bmp
getPixlesFromBMP24 Execution Time:  11 ms
mirrorFilter Execution Time: 4 ms
checkeredFilter Execution Time:  115 ms
diamondFilter Execution Time:  4 ms
writeOutBmp24 Execution Time:  65 ms
Execution Time: 228 ms
```

با توجه به مقدار زمان صرف شده برای اجرای برنامه سری و موازی speedup به صورت زیر خواهد بود.

$$\text{Speedup} = \frac{776}{228} = 3.4$$

همانطور که می‌بینیم مقدار speedup برابر 3.4 است و همانطور که می‌دانیم وقتی از 4 ریسه استفاده می‌کنیم ممکن نیست که بهبود سرعت برنامه 4 برابر شود.