



دانشگاه تربیت مدرس

دانشکده مهندسی برق و کامپیوتر

گزارش سمینار کارشناسی ارشد

مهندسی کامپیوتر (گرایش مهندسی نرم افزار)

گزارش فعالیت پنجم درس سیستم های توزیع شده

دانشجو

علیرضا حیدرآبادی زاده

استاد راهنما

دکتر مریم لطفی

۱. مقدمه:

سیستم‌های توزیع شده و استفاده از پردازش موازی یکی از راه‌حل‌های کارآمد برای بهبود عملکرد در پردازش‌های سنگین هستند. در این فعالیت، هدف مقایسه دو مدل برای پردازش چند ریسمانی (multi-threading) است: یکی با استفاده از پیاده‌سازی ساده `threading` در پایتون و دیگری با استفاده از `ThreadPoolExecutor`.

۲. پیاده‌سازی‌ها:

- در مدل اول، یک سیستم `multi-threaded` ساده با استفاده از ماژول `threading` پیاده‌سازی شد. در این مدل، تعداد زیادی `thread` به صورت دستی ایجاد شده و برای انجام کارهایی مشابه، به صورت همزمان اجرا می‌شوند.
- در مدل دوم، از `ThreadPoolExecutor` برای مدیریت و اجرای تعداد زیادی از `worker`ها استفاده شد. این ابزار به‌طور خودکار تعداد `thread`های فعال را مدیریت کرده و از ایجاد `thread`های اضافی جلوگیری می‌کند.

۳. مقایسه عملکرد:

- در آزمایش‌های انجام شده با تعداد ۱۰۰ `worker`، تفاوت زمان اجرایی بین این دو مدل بسیار کم بود. در مدل ساده، زمان اجرا برابر با ۲.۴۸۶۴۶۳۷۸۵۱۷۱۵۰۹ ثانیه بود و در مدل استفاده از `ThreadPoolExecutor` برابر با ۲.۱۷۷۳۵۰۲۸۲۶۶۹۰۶۷۴ ثانیه.
- این تفاوت‌ها در مقیاس‌های کوچک مانند ۱۰۰ `worker` کمتر قابل توجه هستند. برای مقایسه بهتر، نیاز به آزمایش با تعداد بسیار بیشتری از `worker`ها (مثلاً ۱۰۰۰ یا بیشتر) خواهد بود تا مزیت‌های واقعی `ThreadPoolExecutor` مشخص شوند.
- در ادامه نتیجه آزمایش هر دو مدل را با تعداد ۱۰۰۰۰ `worker` قرار داده شده است.

	Number of Worker	TotalTime (seconds)
Simple Model	100	۲,۴۸۶۴۶۳۷۸۵۱۷۱۵۰۹
ThreadPoolExecutor	100	۲,۱۷۷۳۵۰۲۸۲۶۶۹۰۶۷۴

	Number of Worker	TotalTime (seconds)
Simple Model	10000	۱۶,۶۱۳۸۵۳۲۱۶۱۷۱۲۶۵
ThreadPoolExecutor	10000	۶,۵۳۶۸۰۵۱۵۲۸۹۳۰۶۶

```
Worker 9956 finished.

Total time for 10000 workers: 6.536805152893066 seconds
```

۴. نتیجه گیری:

- در شرایط آزمایش شده با ۱۰۰ worker، هر دو مدل عملکرد مشابهی داشتند. با این حال، ThreadPoolExecutor برای مقیاس‌های بزرگتر می‌تواند به دلیل بهینه‌سازی در مدیریت thread ها و جلوگیری از استفاده بیش از حد منابع، عملکرد بهتری از خود نشان دهد.
- بنابراین، برای مقیاس‌های بزرگتر و سیستم‌هایی که نیاز به مدیریت بهتر منابع دارند، استفاده از سرویس‌های اختصاصی مانند ThreadPoolExecutor مناسب‌تر است.