

در این تمرین یک برنامه‌ی ضرب ماتریس با استفاده از OpenMP به صورت موازی در آمده است.

روش استفاده شده:

روش تجزیه‌ی مساله به این صورت بوده که براساس سطرهای ماتریس حاصل ضرب (ماتریس C) مساله تجزیه شده است. به این صورت که براساس این که می‌خواهیم چند نخ برای توازی داشته باشیم سطرهای ماتریس C را تجزیه می‌کنیم و به این صورت هر کدام از نخ‌ها تعداد مساوی از سطرهای ماتریس C را محاسبه می‌نمایند. با توجه به این که داده‌ها کاملاً محلی است و بر روی یک کامپیوتر اجرا می‌شود، نیازی به ملاحظات مربوط به حداقل کردن داده‌های مورد نیاز برای هر نخ نمی‌باشد (بر خلاف سیستم‌های Message Passing که در آن‌ها این مساله مهم است).

بنابراین تجزیه‌ی براساس سطرهای ماتریس خروجی انتخاب مناسبی به نظر می‌رسد. برای انجام این عمل، حلقه‌ای که در مساله سطرهای ماتریس C را تولید می‌کند، را شکسته ایم. به این صورت که قسمت محاسبات در یک بلاک موازی قرار داده شده و سپس بر اساس id محدوده‌ی حلقه تعیین می‌شود. در واقع هر نخ ابتدا id خود را به دست می‌آورد و سپس با استفاده از آن می‌داند که چه سطرهایی از ماتریس C را باید تولید کند. یعنی مقادیر حد بالا و پایین حلقه را از روی id خود تعیین می‌کند. و به اجرای حلقه در این محدوده می‌پردازد.

با توجه به این که در مرزهای عملیاتی نخ‌ها امکان false sharing وجود دارد، برای از بین بردن این مساله آرایه را بر اساس طول سطرهای حافظه‌ی cache در پردازنده‌ی مورد استفاده، در حافظه align می‌کنیم و نیز برای این که مطمئن شویم که false sharing رخ نخواهد داد، با استفاده از Padding به هر خانه‌ی آرایه‌ی C به اندازه‌ی یک سطر کامل cache حافظه اختصاص می‌دهیم. با توجه به این که در پردازنده‌ی core i7 haswell طول سطرها ۶۴ بایت است، طول PAD باید به اندازه‌ی ۱۶ متغیر integer باشد. بنابراین هنگام تخصیص حافظه به اندازه‌ی ۱۶ برابر حافظه تخصیص می‌دهیم. و هنگام آدرس دهی نیز اندیس تولید شده برای دسترسی به عناصر C را در ۱۶ ضرب می‌کنیم.

نتایج و تحلیل:

# of threads	1 MB	10 MB	100MB	1GB
2	2.108	2.026	?	?
4	3.400	3.536	?	?
8	3.97	4.044	?	?

برای ورودی‌های با سایز 1GB و 100MB به دلیل زمان بسیار زیاد صرف نظر می‌کنیم. چون با توجه به این که با ۱۰ برابر کردن حجم ورودی زمان حدوداً ۳۰ برابر می‌شود و در مورد ورودی ۱۰ مگابایت هر بار اجرای سریال حدوداً یک دقیقه طول می‌کشد. و بنابراین برای مراحل بعد به ترتیب هر بار حداقل نیم‌ساعت و حداقل ۱۵ ساعت طول می‌کشد. ضمن این که در خصوص حافظه نیز ممکن است به مشکل بر بخوریم.

اما نتایج این ورودی‌ها نشان می‌دهد که speedup نسبتاً خوبی به دست آمده است که با بزرگ شدن سایز ورودی‌ها speedup به دست آمده از توازی، بیشتر خود را نشان داده است.