

علیرضا آخوندی

۹۷۳۱۱۰۷

تابع مربوط به این روش در تابع p_prefix_sum آمده است. کد مربوط به شرح زیر است:

```
void p_prefix_sum(int *a, size_t n)
{
    int end[NUM_THREADS];

#pragma omp parallel num_threads(NUM_THREADS)
    {
        int th_num = omp_get_thread_num();

        int inedx = -1;
#pragma omp for schedule(static)
        for (int i = 1; i < n; i++)
        {
            a[i] = a[i] + a[i - 1];
            inedx = i;
        }
        end[th_num] = inedx;

#pragma omp barrier

#pragma omp single
        {
            for (int k = 0; k < NUM_THREADS - 1; k++)
            {
                for (int j = k + 1; j < NUM_THREADS; j++)
                {
                    a[end[j]] += a[end[k]];
                }
            }
        }

#pragma omp for
        for (int k = 0; k < NUM_THREADS - 1; k++)
        {
            int start = end[k] + 1;
            int endd = end[k + 1] - 1;
            int last_val = a[end[k]];
            for (int j = start; j < endd; j++)
                a[j] += last_val;
        }
    }
}
```

کد مربوط به زمان اجرای روش دوم نیز در تابع `h_prefix_sum` آمده است :

```
void h_prefix_sum(int *a, size_t n)
{
    for(long step = 1 ; step < n ; step *= 2){
        int tmp[n];

        #pragma omp parallel num_threads(NUM_THREADS)
        {
            #pragma omp single
            {
                for(int i = 0 ; i < n - step ; i++){
                    #pragma omp task
                    {
                        tmp[i + step] = a[i] + a[i + step];
                    }
                }
            }

            #pragma omp barrier

            #pragma omp for
            for(int i = 0 ; i < n ; i++)
            {
                a[i] = tmp[i];
            }
        }
    }
}
```

زمان‌های اجرای گزارش شده، بعد از ۵ بار اجرا و میانگین گیری به دست آمده است.

زمان اجرای مربوط به سریال به ازای ورودی 10000000 : 0.026690

زمان اجرای مربوط به موازی سازی روش اول به ازای ورودی 10000000 و 8 هسته : 0.011595

زمان اجرای مربوط به موازی سازی روش اول به ازای ورودی 1000000
و 8 هسته : 22.467438

دلیل ضعیف کار کردن الگوریتم دوم تعداد زیاد تسک ها و overhead بالای مربوط به context switch است.

در صورتی که از gpu استفاده کنیم چون در آن از نخ های سبک تری استفاده می کنیم که هزینه context switch در آنها روش دوم به ما تسریع بهتری می دهد.