

## امتحان پایانی معماری سیستم‌های موازی

مجموع نمره‌ها: ۱۰۰

زمان آزمون: ۹۰ دقیقه

۱ (۳۰) در قطعه کدهای زیر، عدد  $x$  در بین تعدادی عدد جستجو می‌شود. در یکی از این قطعه کدها از آرایه و در دیگری از لیست پیوندی برای نگهداری عددها استفاده شده است. الف) محلی‌گرایی (Locality) را در این دو قطعه کد مقایسه نمایید. ب) چگونه و در چه صورتی کامپایلر می‌تواند به کاهش تأخیر دسترسی به حافظه در لیست پیوندی کمک کند؟ ج) با دلیل پیش‌بینی کنید که در هر یک از این دو قطعه کد، آیا استفاده از ریسمان‌های سخت‌افزاری (Threading)، پردازنده‌های چند هسته‌ای یا اجرای دستورات به صورت SIMD در افزایش سرعت مؤثر است یا خیر (در صورت امکان می‌توان دورهای حلقه‌ها را بین چند ریسمان سیستم عامل تقسیم کرد). هر یک از این سه عامل را به صورت مجزا بررسی کنید.

```
# Array                                # Linked list
for (i = 0; i < N; i++)                while (cur) {
    if (A[i] == x)                      if (cur->val == x)
        return 1;                       return 1;
return 0;                               cur = cur->next;
                                        }
                                        return 0;
```

۲ (۳۰) در قطعه کد زیر، آرایه‌ی  $A$ ،  $N$  عدد را نگه می‌دارد، آرایه‌ی  $P$  جایگشتی از اعداد صحیح مثبت کوچک‌تر از  $N$  است (یعنی اعداد صفر تا  $N - 1$  به ترتیبی نامشخص در آرایه‌ی  $P$  قرار دارند) و آرایه‌ی  $B$  خروجی است (تابع  $compute()$  مقداری را از روی عدد ورودی محاسبه می‌کند و بر می‌گرداند). در مدل حافظه‌ی مشترک، برای موازی کردن این قطعه کد، دورهای حلقه بین پردازنده‌ها تقسیم شده‌اند. چگونه اشتراک کاذب (False sharing) رخ می‌دهد و چگونه می‌توان بدون تغییر آرایه‌ی  $B$  از آن جلوگیری نمود؟

```
for (i = 0; i < N; i++)
    B[A[P[i]]] = compute(A[i]);
```

۳ (۳۰) آیا امکان دارد در یک حافظه‌ی Coherent پس از اجرای موازی سه قطعه کد زیر توسط سه پردازنده، مقدار نهایی متغیرهای  $a$  و  $b$  برابر یک شود (مقدار اولیه‌ی متغیرها صفر است)؟ اگر خیر، کدام شرط لازم Coherency و چگونه نقض می‌شود؟

```
# Processor #1                # Processor #2                # Processor #3
if (a == 0) {                 if (a == 1) {                 if (a == 2) {
    a = 1;                    a = 2;                        a = 3;
}                              b = 1;
                                }
                                }
```

۴ (۳۰) در پروتکل MESI، دنباله‌ی پیغام‌هایی که توسط حافظه‌ی نهان پردازنده‌ها روی گذرگاه مشترک (Bus) قرار می‌گیرند تا در قطعه کد سؤال قبل مقدار متغیر  $a$  و  $b$  به ترتیب دو و یک شود را بنویسید. یادآوری می‌شود که پیغام‌های گذرگاه در MESI عبارتند از BusRd، BusRdX، flush و (BusWB) (فرض کنید حافظه‌های نهان Write-allocate هستند).