

## پروژه برنامه‌نویسی نهایی اسembلی (۵ نمره از کل + حداکثر یک نمره امتیازی)

فقط یکی از دو گزینه پروژه الف یا ب را برگزینید:

### الف: پروژه نوع اول: برنامه‌نویسی اسembلی x86

این پروژه سه مرحله دارد که سومی امتیازی است (تا ۱ نمره). هدف آن بکارگیری دستورهای اسembلی، به طور خاص، دستورهای برداری (SIMD) مثل:

- 64-bit: MMX
- 128-bit: SSE, SSE2, SSE3, SSE4, SSE4.1, SSE4.2 (streaming SIMD extension)
- 256-bit: AVX, AVX2 (advanced vector extension)
- 512-bit: AVX-512

برای پردازنده‌های اینتل و AMD و دستورهای NEON برای پردازنده‌های جدیدتر ARM، در یک کاربرد پردازش تصویر و شناسایی الگو در تصاویر است.

البته امروزه از این نوع دستورها چه مستقیماً و چه به کمک دستورهای معادل سطح بالا (مثل intrinsic functions x86<sup>۱</sup> اینتل)، که با اضافه کردن <x86intrin.h> به برنامه، به کامپایلر اجازه ترجمه دستورهایی مشابه زیر را در زبان-های سطح بالا می‌دهد

```
__m256 * ma = (__m256 *) (&a[i*width]);  
__m256 * mb = (__m256 *) (&b[i*width]);  
__m256 * mr = (__m256 *) (&r[i*width]);  
*mr = _mm256_mul_ps(*ma, *mb);
```

یا کتابخانه‌های آماده و کامپایلرهای مناسب در کاربردهای هوش مصنوعی و یادگیری ماشین نیز استفاده می‌شود. بنابراین هدف این است که یک کار عملی جذاب و مفید را به کمک دستورهای اسembلی تجربه کنید و از سرعت اجرای آن لذت ببرید. توصیه می‌شود از این فرصت برنامه‌نویسی برای صداردن توابع مفید زبان‌های سطح بالا از داخل برنامه اسembلی و یا بالعکس استفاده کنید تا در آینده بتوانید برنامه‌های خاص اسembلی را داخل برنامه‌های دیگر درج کنید.

### ۱- طراحی برنامه Convolution

۱-۱ یک برنامه اسembلی Convolution-2D بنویسید که به کمک یک برنامه سطح بالای دیگر (مثل C یا Python) یک تصویر را از ورودی گرفته و با اعمال تابع Convolution یا فیلتر دلخواه، آن را به یک تصویر دیگر تبدیل کند (مثلاً برای بهبود کیفیت تصویر یا تشخیص لبه‌ها...). قسمت ورود و خروج اطلاعات (تصویر) یا اندازه‌گیری زمان را می‌توانید با زبان سطح بالا انجام دهید و از صحت نتایج نیز مطمئن شوید.

۲-۱ زمان اجرای برنامه کامل را با برنامه‌های مشابه موجود یا برنامه خودتان به دست آورید.

<sup>۱</sup> <https://www.intel.com/content/www/us/en/docs/intrinsics-guide/index.html#>

۱- همین طور مقدار تسريع (Speedup)، يعني نسبت زمان اجرای برنامه فقط C یا فقط پایتون معمولی را به زمان اجرای جدید، که با کمک دستورهای اسمبلی برای جایگزینی قسمت‌های داغ یا زمانبر برنامه پیاده شده است، مقایسه کنید. دقت کنید که گاه کامپایلرها خود به شدت برنامه را بهینه ترجمه و چه بسا از دستورهای SIMD نیز استفاده می‌کنند

توضیح اینکه عمل Convolution (به زبان ساده) یک تابع را بر روی محور افقی از روی یک تابع دیگر عبور می‌دهد و در هر نقطه برخورد (یا در نقاط گستته) حاصل ضرب این دو تابع را محاسبه و ثبت می‌کند. (تعداد قابل توجهی از پردازش‌های تصویر و مدل‌های یادگیری ماشین، بر اساس تابع Convolution پیاده‌سازی شده است). می‌توانید توابع را با ماتریس ۵ در ۵ و ۳ در ۳ پیاده کنید.

- توجه کنید که عمل Convolution خود مبتنی بر ضرب ماتریس است<sup>۲</sup> که براحتی با دستورهای SIMD پیاده‌سازی می‌شود.

۲- از برنامه و تجربه قبلی پیاده‌سازی تابع Convolution، برای پیاده‌سازی یک برنامه شناسایی الگو<sup>۳</sup> یا شیء استفاده و با انجام چندین آزمایش روی تصاویر متعدد، دقت تشخیص را محاسبه کرده، تسريع را بر اساس اجرا روی بیش از صد تصویر متوالی روی GPU و نه CPU، به دست آورید. (برای انتخاب الگوریتم تشخیص الگو، کافی است فیلتر مناسب را برای شناسایی هدف یا شیء مورد نظر در تصویر درست انتخاب کنید)

۳- (امتیازی) برنامه تشخیص توده بیماری را مبتنی بر شبکه پیچشی (CNN) با دستورهای اسمبلی و زبان سطح بالا بنویسید و ارزیابی کنید. می‌توانید از دادگان (Dataset) تصاویر پزشکی موجود در اینترنت استفاده کنید.

## ب- پروژه نوع دوم: کار با پردازنده ARM

یک برنامه اسمبلی روی گوشی خود (به شرط اینکه مبتنی بر معماری ARM باشد) بنویسید که همزمان چهار تا توپ یا هدف متحرک با سرعت قابل تنظیم شلیک کند و شما بتوانید توپ‌ها را بگیرید یا منهدم کنید.

یا اینکه برنامه‌ای روی گوشی بنویسید که همزمان به چند نفر بتواند پیامک بفرستد.

### توجه:

۱- به غیر از تهیه گزارش برنامه اصلی، درج کد قابل اجرا و شرح روش انجام کار و عکس‌هایی از اجرای برنامه، تحويل عملی پروژه به صورت حضوری (و در صورت مجازی بودن ابتدای نیمسال بعد، به صورت غیر حضوری برای دانشجویان مقیم خارج از تهران) خواهد بود.

۲- این تحويل گیری می‌تواند با درخواست شما و تعیین قرار قبلی باشد و یا مطابق یک زمانبندی و تنظیم وقت از پیش اعلام شده قبلاً از ۱ اسفند ۱۴۰۴.

۳- از آنجا که استفاده از ابزارهای هوش مصنوعی یا کدهای آماده را محتمل می‌دانم، باید بر هر خط برنامه خود مسلط و به طرز کار یا عملکرد آن آشنا باشید و گرنه هیچ نمره‌ای نخواهید گرفت. در ضمن هر خط برنامه اسمبلی شما و توابع مهم زبان سطح بالا و شیوه صدا زدن توابع از اسمبلی یا بالعکس باید یک توضیح به زبان فارسی داشته باشد.

<sup>2</sup> [https://en.wikipedia.org/wiki/Convolution#/media/File:Convolution\\_of\\_spiky\\_function\\_with\\_box2.gif](https://en.wikipedia.org/wiki/Convolution#/media/File:Convolution_of_spiky_function_with_box2.gif)  
[https://en.wikipedia.org/wiki/Kernel\\_\(image\\_processing](https://en.wikipedia.org/wiki/Kernel_(image_processing)

<sup>3</sup> Pattern recognition