

# گزارش پروژه‌ی برنامه‌نویسی چندهسته‌ای

محمدامین محمدی - ۹۴۳۱۰۲۰

## مقدمه

در این پروژه باید چند نمونه‌ی آهنگ و چند آهنگ را می‌گرفتیم و سپس تعیین می‌کردیم که هر یک از نمونه‌های دریافتی جدا شده از کدام آهنگ هستند. برای این منظور، برنامه‌ی ما باید در ورودی آدرس دو پوشه، یکی پوشه‌ی حاوی نمونه‌ها و دیگری پوشه‌ی حاوی آهنگ‌های اصلی را بگیرد و با اجرای کد مناسب، تصمیم نهایی را اعلام کند. برای این منظور می‌توان از تبدیل فوریه و روش‌های موازی‌سازی روی GPU برای تسریع این فرآیند استفاده کرد.

## روش کار

### ۱. خواندن ورودی

برای انجام این پروژه، ابتدا با کمک کتابخانه‌ی `dirent`، فایل‌های پوشه‌های مختلف را می‌یابیم. سپس با استفاده از توابع `fopen` و `fgetc` شروع به خواندن فایل‌ها و ذخیره‌ی اطلاعات آن می‌کنیم (برای سرعت بیشتر از همین توابع اولیه استفاده می‌کنیم، نه توابع `high level` در `C++`). عملیات خواندن فایل‌های درون هر پوشه با استفاده از `OpenMP` تسریع می‌شود تا فایل‌ها به صورت موازی خوانده شوند.

### ۲. گرفتن تبدیل فوریه

پس از دریافت و تبدیل ورودی‌ها به فرمت خواسته‌شده، از کتابخانه‌ی `CuFFT` برای گرفتن تبدیل فوریه از داده‌های ورودی (نمونه‌ها) با استفاده از قابلیت‌های GPUهای `Cuda` استفاده می‌کنیم. در صورت عدم استفاده از تبدیل فوریه، مقایسه‌ی آهنگ و نمونه بسیار زیاد طول خواهد کشید.

### ۳. سردادن پنجره‌ای روی آهنگ و مقایسه

سپس یک پنجره به اندازه‌ی نمونه‌ی ورودی روی آهنگ مورد مقایسه در نظر می‌گیریم و با سردادن آن با پارامتر پنجره‌ی مناسب، از آن پنجره‌ی آهنگ تبدیل فوریه می‌گیریم و تبدیل فوریه‌ی این پنجره را با تبدیل فوریه‌ی نمونه‌ی موردنظر مقایسه می‌کنیم. برای مقایسه، از دو معیار `Cosine Similarity` و `LAD Similarity` به صورت موازی‌شده روی GPU استفاده کردیم. این کار نیاز به یک الگوریتم `Reduction` روی آرایه‌ای با اندازه‌ی دلخواه داشت که این الگوریتم را در تمرین ۵ پیاده کرده بودیم.

### ۴. انتخاب آهنگ نهایی

در نهایت، آهنگی که طبق معیار استفاده شده (هر دو پیاده شده‌اند) شبیه‌ترین به نمونه‌ی ورودی باشد، به عنوان خروجی آن نمونه انتخاب می‌شود.