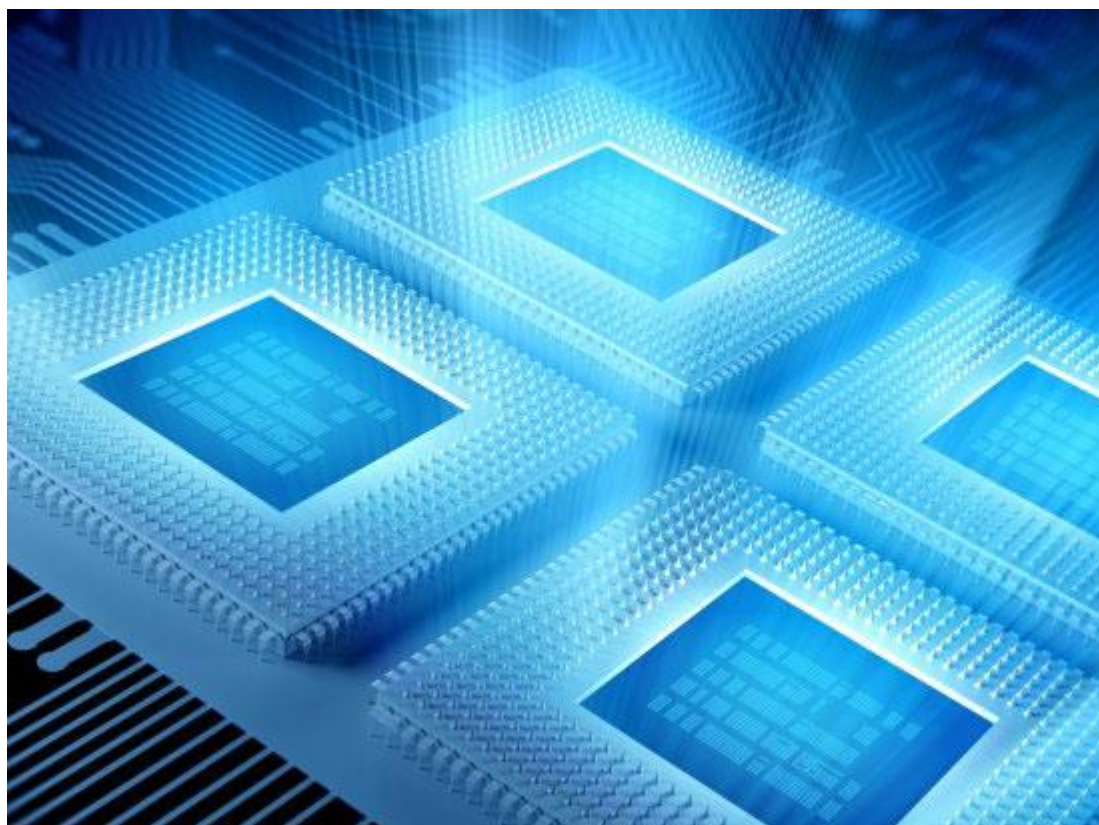




آموزش نصب پیش نیازها و اجرای پروژه Gaussian blur filter

OpenMP



محمدحافظ یاری

Yari@mail.um.ac.ir

دانشگاه فردوسی مشهد

دی ۱۴۰۰

به نام خدا

۱. مقدمه

فیلتر Gaussian blur^۱، یک فیلتر در مبحث پردازش تصویر است که با اعمال تابع گوسین بر روی تصویر آن را تار می‌کند. چون این عمل بر روی هر یک از پیکسل‌های تصویر به صورت جداگانه انجام می‌شود برای موازی‌سازی بسیار مناسب است. در این فایل آموزشی، نحوه نصب کامپایلر mingw بر روی سیستم عامل ویندوز ۶۴ بیتی و همچنین پیش‌نیازهای کامپایلر نسخه OpenMP و اجرای پروژه Gaussian blur filter بررسی می‌شود. این پروژه با ساده‌ترین روش ممکن به زبان C++ پیاده‌سازی شده است.

۲. ساختار پروژه

کد سرچال پروژه با زبان C++ از لینک زیر قابل دریافت است.

https://github.com/m-hafez/FUM_MP_ParallelProject-1_1400-1401

📁	CudaCode
📁	cimg
📁	img
📄	.gitignore
📄	README.md
📄	main.cpp

• پوشه CudaCode

زیرساخت پروژه با CUDA است، برای این آموزش نیازی به آن نیست و می‌توانید آن را حذف کنید.

• پوشه cimg

این پوشه شامل کتابخانه CImg^۲ برای خواندن و تبدیل کردن تصویر ورودی به آرایه است تا بتوان با ساده‌ترین روش فیلتر را بر روی آن اعمال کرد.

• پوشه img

فایل تصویر ورودی برنامه در این پوشه بانام input.jpg قرار دارد.

• فایل main.cpp

تمام کد پروژه در این فایل قرار دارد که شامل توابع زیر است. برای کامپایل پروژه باید این فایل کامپایل شود؛ که در قسمت ۴ توضیح داده شده است.

^۱ https://en.wikipedia.org/wiki/Gaussian_blur

^۲ <https://cimg.eu/>

```
• double **getGaussian(int height, int width, double sigma)
```

این تابع یک فیلتر به اندازه $\text{height} \times \text{width}$ می‌سازد و برمی‌گرداند.

```
• double ***applyFilter(double ***image, double **filter, int width, int height, int filterWidth, int filterHeight)
```

این تابع فیلتر ساخته‌شده توسط تابع `getGaussian`، `(double **filter)` را بر روی تصویر `(double ***image)` اعمال می‌کند و تصویر تار شده را برمی‌گرداند؛ که این عمل در تابع `main` در خط زیر انجام می‌شود.

```
kernel = applyFilter(kernel, getGaussian(filterHeight, filterWidth, 10.0), width, height, filterWidth, filterHeight);
```

اکنون تصویر تار شده در `kernel` قرار دارد.

۳. پیش‌نیازها

۱-۱- کامپایلر MinGW

- لینک دانلود کامپایلر MinGW نسخه ۶۴ بیتی برای ویندوز:

https://iweb.dl.sourceforge.net/project/mingw-w64/Toolchains%20targetting%20Win64/Personal%20Builds/mingw-builds/8.1.0/threads-win32/seh/x86_64-8.1.0-release-win32-seh-rt_v6-rev0.7z

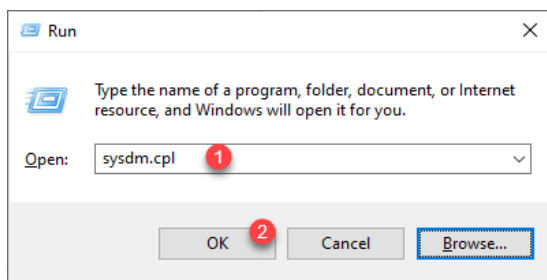
- نحوه نصب

بعد از دانلود کامپایلر از لینک بالا، فایل `x86_64-8.1.0-release-win32-seh-rt_v6-rev0.7z` را در اختیاردارید. کافی است محتویات فایل فشرده که پوشه `mingw64` است را در مکان دلخواهی قرار دهید. مثلاً در درایو C. سپس مسیر پوشه `bin` را باید به `path variable` های ویندوز اضافه کنید. با توجه که در این آموزش، پوشه `mingw64` در درایو C قرار گرفته، پس مسیر پوشه `bin` به صورت زیر است.

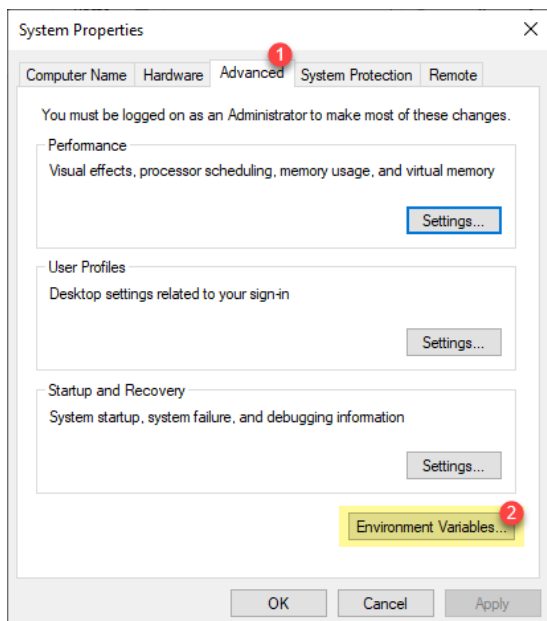
`C:\mingw64\bin`

برای اضافه کردن این مسیر به `path variable` های ویندوز، با فشردن کلیدهای ترکیبی `Windows + R`،

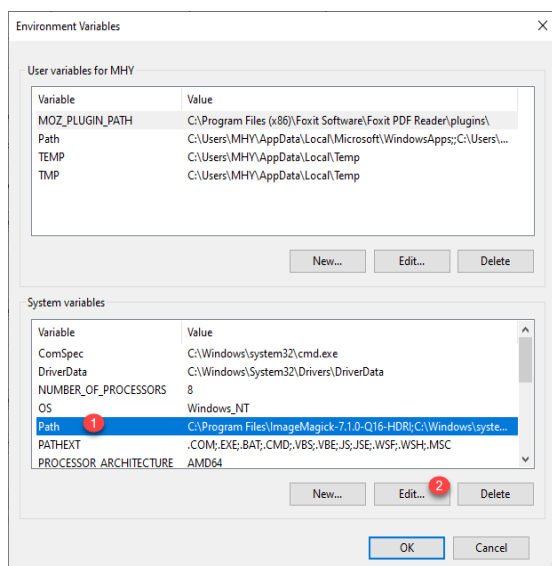




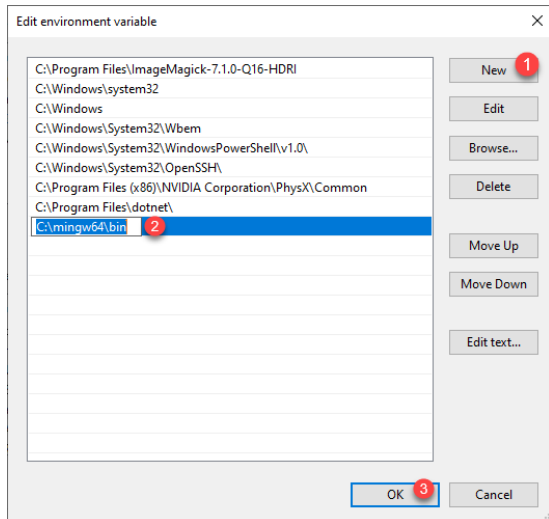
پنجره Run باز شده و با وارد کردن sysdm.cpl



پنجره System Properties باز می‌شود. سپس از سربرگ Advanced، بر روی Environment Variables... کلیک کنید.

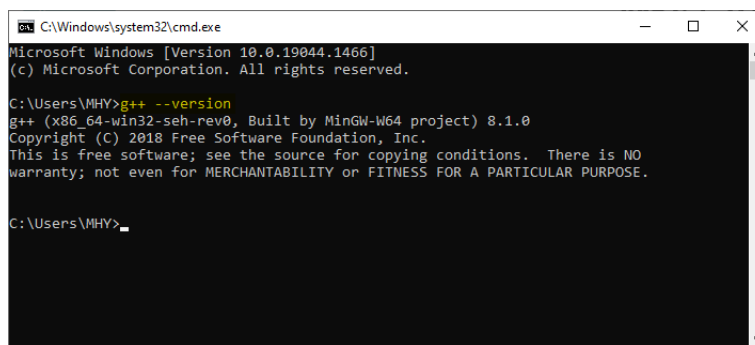


سپس در پنجره Environment Variables در قسمت System variables، Path را انتخاب کرده و بر روی دکمه Edit... کلیک کنید.



در پنجره Edit environment variable دکمه New را زده و در کادر ایجادشده مسیر پوشه bin را وارد کنید؛ و در انتها بر روی Ok کلیک کنید؛ و به همین ترتیب برای پنجره‌های قبلی بازشده، بر روی Ok کلیک کنید.

اکنون کامپایلر mingw بر روی سیستم شما نصب‌شده است و می‌توانید از آن استفاده کنید. برای اطمینان از نصب صحیح آن، command prompt ویندوز را با وارد کردن دستور cmd در پنجره Run، باز کنید؛ و دستور ++ --version را در آن وارد کنید. این دستور نسخه کامپایلر mingw را نمایش می‌دهد و باید خروجی مانند تصویر زیر باشد.



۲-۱- ImageMagick

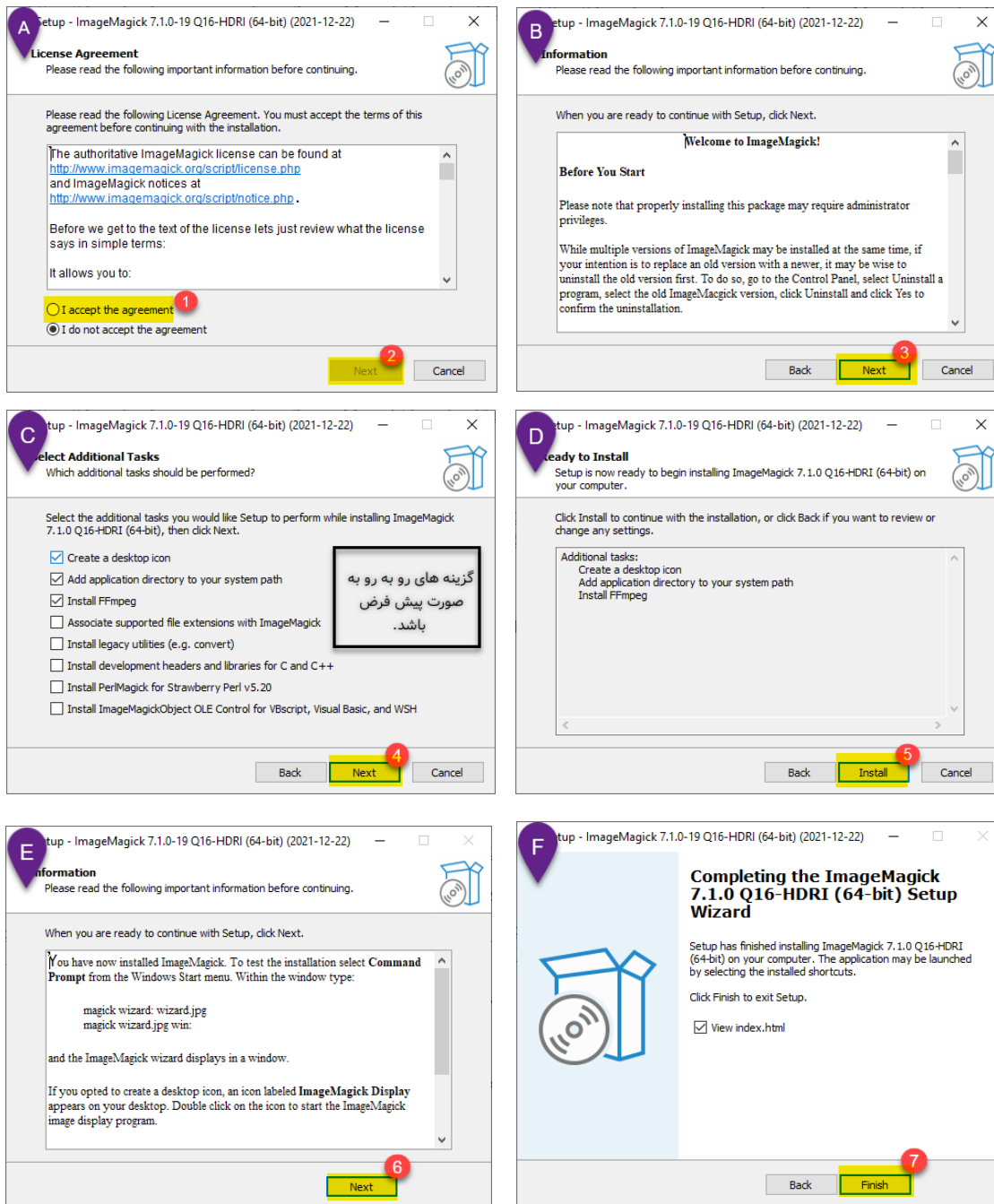
از ImageMagick فقط برای نمایش تصویر توسط کد استفاده می‌شود.

- لینک دانلود ImageMagick برای ویندوز ۶۴ بیتی:

<https://imagemagick.org/script/download.php#windows>

- نحوه نصب

بعد از دانلود از لینک بالا فایل ImageMagick-...-x64-dll.exe را در اختیار دارید. برای نصب کافی است تا فایل را اجرا کرده و مراحل مشخص‌شده در تصاویر زیر را دنبال کنید.



۴. کامپایل و اجرا

برای کامپایل پروژه توسط کامپایلر mingw و استفاده از OpenMP از دستور زیر در cmd باز شده در پوشه پروژه، استفاده کنید.

```
g++ .\main.cpp -o .\main -O2 -lgdi32 -fopenmp
```

در انتها فایل اجرایی برنامه بانام main.exe در پوشه پروژه ایجاد می‌شود.