Setting Up MLX42

updated: 18/05/2022

During the Spark Session, we'll be writing a little program that can open a window, draw some pixels, and even move pixels!

Before we can do that, we need to set up the MLX42 library and **link** the compiled library with our source code.

1. Download the MLX42 library into the **root directory**.
   - Download here
   - **For the next steps we will assume you've called the folders `MLX42` or `MLX`.**
2. Using MLX42 requires that we link the necessary **internal API's**.
   - For macOS: Install Homebrew42. Then run the following in the terminal:

     ```
     brew update
     brew install glfw
     ```

   - For Linux: Simply run this to install the required dependencies:

     ```
     sudo apt update
     sudo apt install build-essential libx11-dev libglfw3-dev libglfw3 xorg-dev
     ```

   - For Windows: Please refer to the installation guide on github.
3. Create a `main.c` in the root directory with the following content (remember to include **MLX42/MLX42.h**):

   ```
   int    main(void)
   {
       mlx_t   *mlx;mlx = mlx_init(256, 256, "Hello World!", false);
   return (0);}
   ```

   Do not worry yet what the parameters are, we will cover them later.
   This will help you check at the end if you're linking your MLX42 correctly to your source files.
4. Now let's create our own Makefile in the root directory, which will compile our `main.c` into a program. Add the required rules - `$(NAME)`, `clean`, `fclean`, `re`, `all`.

Here's a helpful guide on Makefiles written by another student, Noah Loomans.

5. Compile the MLX42 library, so that you get a `libmlx42.a` file.
   Tip: You could have your Makefile do all this too. *Hint:* `make -C dir`

6. Now we finally come to combining everything together and link everything properly.

   - For macOS:

   ```
   $(NAME): $(OBJ)
       $(CC) $(OBJ) libmlx42.a -lglfw -L "/Users/$USER/.brew/opt/glfw/lib/" -o $(NAME
   ```

   - For Linux:

   ```
   $(NAME): $(OBJ)
       $(CC) $(OBJ) libmlx42.a -ldl -lglfw -o $(NAME)
   ```

   - For Windows:

   ```
   $(NAME): $(OBJ)
       $(CC) $(OBJ) libmlx42.a -lglfw3 -lopengl32 -lgdi32 -o $(NAME)
   ```

   Tip: Don't forget to define the header location. *Hint:* `-I <Directory>`

7. Now try running `make` in your project directory. Does your program compile without errors? Your program itself won't do anything for now. If everything works, great! Now you're ready for the SparkSession and your project!