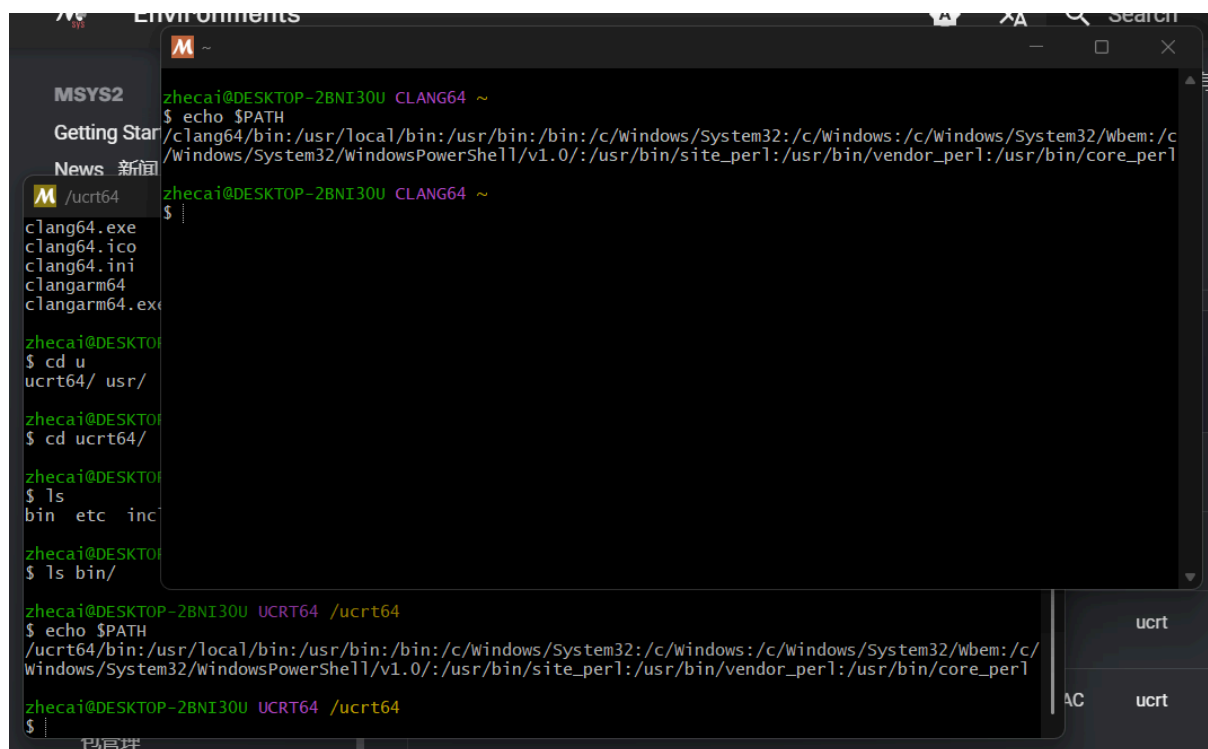
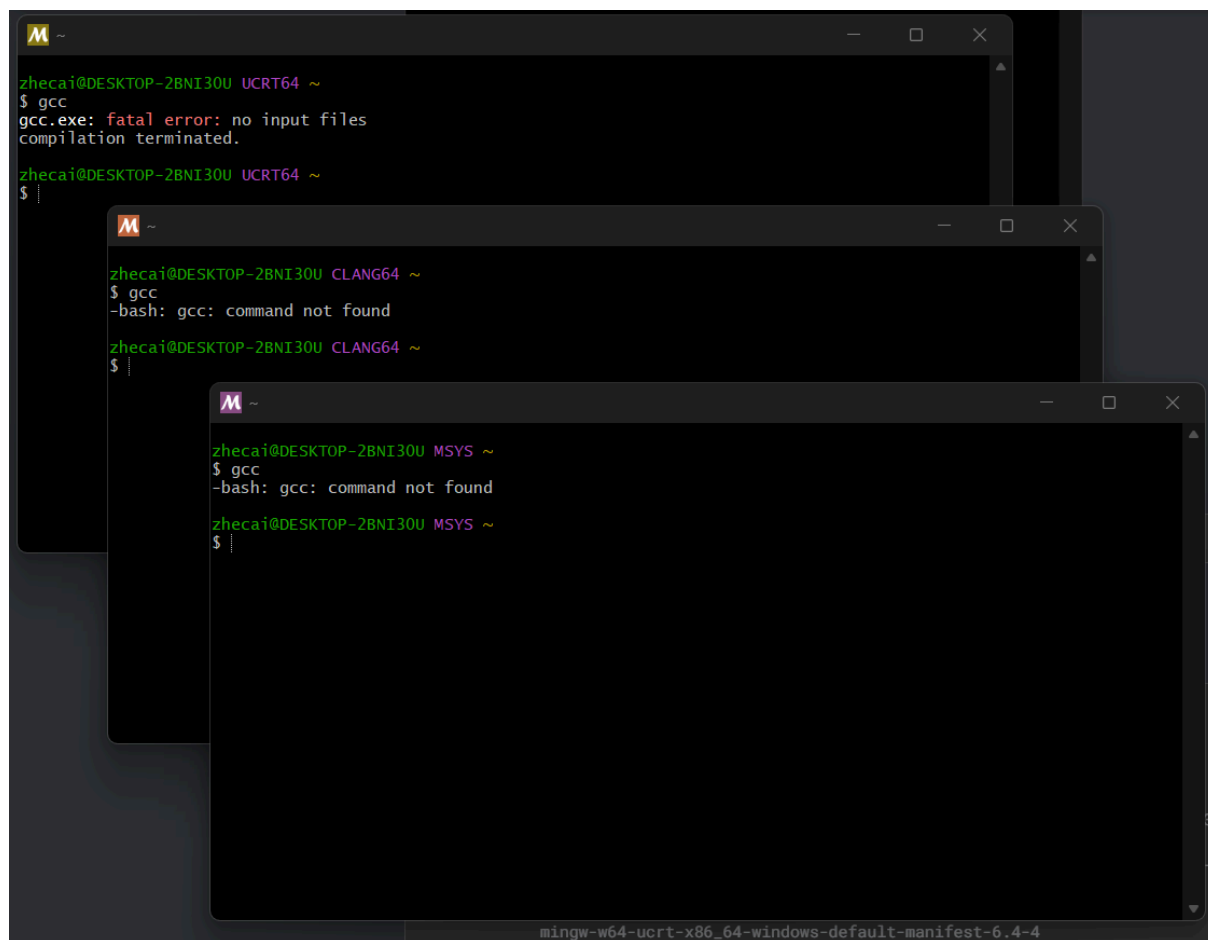


安装 msys2 之后给了几个不同的 bash 环境



```
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$ echo $PATH  
/usr/local/bin:/usr/bin:/bin:/c/Windows/System32:/c/Windows:/c/Windows/System32/Wbem:/c/Windows/System32/WindowsPowerShell/v1.0:/usr/bin/site_perl:/usr/bin/vendor_perl:/usr/bin/core_perl  
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$ cd /usr/  
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$ ls  
bin etc include  
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$ ls bin/  
zhecai@DESKTOP-2BNI30U UCRT64 /usr64  
$ echo $PATH  
/usr64/bin:/usr/local/bin:/usr/bin:/bin:/c/Windows/System32:/c/Windows:/c/Windows/System32/Wbem:/c/Windows/System32/WindowsPowerShell/v1.0:/usr/bin/site_perl:/usr/bin/vendor_perl:/usr/bin/core_perl  
zhecai@DESKTOP-2BNI30U UCRT64 /usr64  
$
```

pacman 安装时可以特地只为一个环境安装工具



```
zhecai@DESKTOP-2BNI30U UCRT64 ~  
$ gcc  
gcc.exe: fatal error: no input files  
compilation terminated.  
zhecai@DESKTOP-2BNI30U UCRT64 ~  
$  
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$ gcc  
-bash: gcc: command not found  
zhecai@DESKTOP-2BNI30U CLANG64 ~  
$  
zhecai@DESKTOP-2BNI30U MSYS ~  
$ gcc  
-bash: gcc: command not found  
zhecai@DESKTOP-2BNI30U MSYS ~  
$
```

来到重点

```
zhecai@DESKTOP-2BNI30U UCRT64 ~
$ ls /ucrt64/bin/ | grep make
cmake.exe
mingw32-make.exe
psl-make-dafsa

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ ls /usr/bin/ | grep make
make.exe
makepkg
makepkg-mingw
makepkg-template
psl-make-dafsa

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ mingw32-make
mingw32-make: *** No targets specified and no makefile found. Stop.

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ make
make: *** No targets specified and no makefile found. Stop.

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ which make
/usr/bin/make

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ which cmake
/ucrt64/bin/cmake

zhecai@DESKTOP-2BNI30U UCRT64 ~
$ where cmake
C:\msys64\ucrt64\bin\cmake.exe

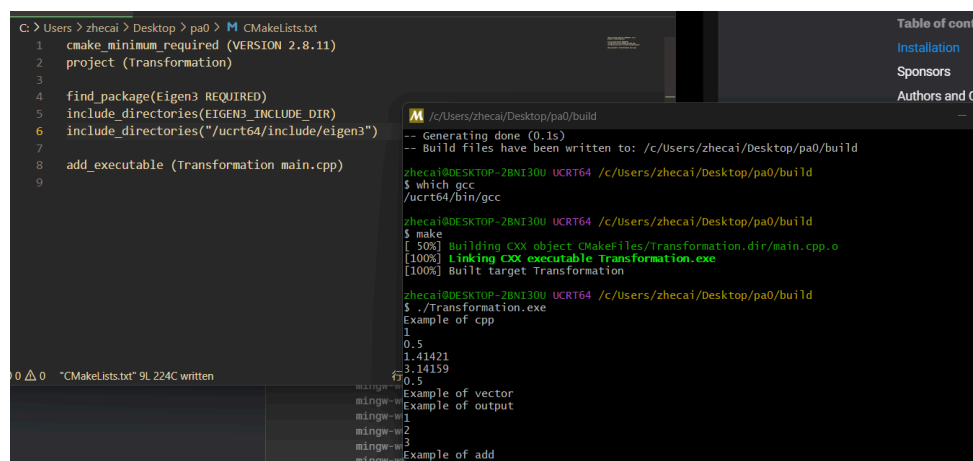
zhecai@DESKTOP-2BNI30U UCRT64 ~
$
```

只为 ucrt64 安装的 make 叫做 mingw32-make，但可以直接 `pacman -S make` 安装 make 在 /usr/bin，也就是所有 msys 环境的公共环境。

(话说还能直接在 bash 里用 `cmd` 命令哦，好牛逼

顺便 ucrt 的 cmake 默认是不生成 makefile 的，msys 自带的才生成，我也懒得配置了，make 和 cmake 都用 msys 的基环境的就行了。

eigen3 只为 mingw64 ucrt64 clang64 这种子环境编译了，所以要在 cmake 里引用一下。



```
C:\Users\zhecai\Desktop\pa0> M CMakeLists.txt
1 cmake_minimum_required (VERSION 2.8.11)
2 project (Transformation)
3
4 find_package(Eigen3 REQUIRED)
5 include_directories(EIGEN3_INCLUDE_DIR)
6 include_directories("/ucrt64/include/eigen3")
7
8 add_executable (Transformation main.cpp)
9

C:\Users\zhecai\Desktop\pa0\build>
-- Generating done (0.1s)
-- Build files have been written to: C:/Users/zhecai/Desktop/pa0/build

zhecai@DESKTOP-2BNI30U UCRT64 /c:/Users/zhecai/Desktop/pa0/build
$ which gcc
/ucrt64/bin/gcc

zhecai@DESKTOP-2BNI30U UCRT64 /c:/Users/zhecai/Desktop/pa0/build
$ make
[ 50%] Building CXX object CMakeFiles/Transformation.dir/main.cpp.o
[100%] Linking CXX executable Transformation.exe
[100%] Built target Transformation

zhecai@DESKTOP-2BNI30U UCRT64 /c:/Users/zhecai/Desktop/pa0/build
$ ./Transformation.exe
Example of cpp
1
0.5
1.41421
2.14159
Example of vector
Example of output
mingw-w64
mingw-w64
mingw-w64
Example of add
```