

Getting Started with LLVM for Visual Studio

Homepage: <https://sites.google.com/site/doc4code/>

Email: sj6219@hotmail.com

2020/4/25

- LLVM을 checkout한다.
 - > git clone https://github.com/llvm/llvm-project.git
- x64 Native Tools Command Prompt for VS 2019 를 이용하여 LLVM.sln 파일을 생성한다.

```
> cd llvm-project
> mkdir build
> cd build
> cmake -DLLVM_ENABLE_PROJECTS=clang -G "Visual Studio 16 2019" -A x64
-Thost=x64 ../llvm
```

- 위에서 생성된 llvm-project\build\LLVM.sln를 Visual Studio 프로그램으로 연다.
clang 프로젝트 등을 빌드한다.
- llvm-project\build\Debug\bin\hello.c를 생성한다.

```
#include <stdio.h>

int main(int argc, char **argv)
{
    printf("hello world\n");
    return 0;
}
```

- x64 Native Tools Command Prompt for VS 2019를 실행한다.
 - > cd llvm-project\build\Debug\bin

- C 파일을 컴파일 해서 LLVM 중간 코드를 생성한다.
> clang -c hello.c -emit-llvm -o hello.bc
- 생성된 중간 코드를 실행해본다.
> lli hello.bc
hello world
- 중간 코드를 disassemble해서 LLVM assembly 코드를 살펴 본다.
> llvmdis < hello.bc
- 중간코드를 obj 파일로 만든다.
> llc -filetype=obj hello.bc
- Microsoft의 Link를 이용하여 실행 파일을 만든다.
> link hello.obj -defaultlib:libcmt
- 실행해 본다.
> hello.exe
hello world
- Clang을 이용해 직접 실행파일을 만들수도 있다.
> clang hello.c -o hello.exe
- Kaleidoscope-ch4 프로젝트를 테스트해보자.
ready> **def test(n) n+1;**
ready> Read function definition:define double @test(double %n) {
entry:
%addtmp = fadd double %n, 1.000000e+00
ret double %addtmp
}

ready> **test(3);**
ready> Evaluated to 4.000000

