#include <stdio.h>

#include <dlfcn.h>

int main(int argc, char\* argv[]) {

int i = 10;

void (\*myfunc)(int \*); void \*dl\_handle;

char \*error;

dl\_handle = **dlopen**("libmymath.so", RTLD\_LAZY);//RTLD\_NOW

if(!dl\_handle) {

printf("dlopen() error - %s\n", **dlerror**()); return 1;

}

//Calling mul5(&i);

myfunc = **dlsym**(dl\_handle, "mul5"); error = **dlerror**();

if(error != NULL) {

printf("dlsym mul5 error - %s\n", error); return 1;

}

myfunc(&i);

//Calling add1(&i);

myfunc = **dlsym**(dl\_handle, "add1"); error = **dlerror**();

if(error != NULL) {

printf("dlsym add1 error - %s\n", error); return 1;

}

myfunc(&i);

printf("i = %d\n", i);

**dlclose**(dl\_handle);

return 0;

OPTIMIZE = -O2

CC = gcc

CFLAGS = $(OPTIMIZE) -g3 -Wall -Wextra -march=native -mtune=native -mrdrnd

randmain:

$(CC) $(CFLAGS)-c randmain.c -o randmain.o

$(CC) $(CFLAGS) -c randcpuid.c -o randcpuid.o

$(CC) $(CFLAGS) -ldl -Wl,-rpath=$(PWD) randmain.o randcpuid.o -o randmain\

randlibhw.so:

$(CC) $(CFLAGS) -c -fPIC randlibhw.c -o randlibhw.o

$(CC) $(CFLAGS) -shared -fPIC -o randlibhw.so randlibhw.o

randlibsw.so:

$(CC) $(CFLAGS) -c -fPIC randlibsw.c -o randlibsw.o

$(CC) $(CFLAGS) -shared -fPIC -o randlibsw.so randlibsw.o