```c
#include <stdio.h>

extern double sqrt(double x) __attribute__ ((weak));

void func()
{
    if(sqrt)
        printf("%f\n", sqrt(10.0));
    else
        printf("no sqrt\n");
}

int main()
{
    func();
    return 0;
}
```

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ gcc -o weak weak.c
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ ./weak
no sqrt
```

可以理解

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ gcc -o weak weak.c -lm
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ ./weak
no sqrt
```

这是为什么?

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ cat weak.c
#include <stdio.h>

extern double sqrt(double x) __attribute__ ((weak));

void func()
{
    if(sqrt)
        printf("%f\n", sqrt(10.0));
    else
        printf("no sqrt\n");
}

walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ gcc -fPIC -shared -o we
ak.so weak.c -g
```

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ cat weak_main.c
#include <stdio.h>

double sqrt(double x)
{
    printf("sqrt ok\n");
    return  1.0;
}

int main()
{
    func();
    return 0;
}

walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ gcc weak_main.c ./weak.so -g
```

运行结果

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ ./a.out
3.162278
```

太奇怪了!!!

```
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ cat weak_main.c
#include <stdio.h>

//double sqrt(double x)
//{
//  printf("sqrt ok\n");
//  return  1.0;
//}

int main()
{
    func();
    return 0;
}

walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ gcc weak_main.c ./weak.so -g
walterzh@walterzh-Precision-T1650:~/work2/temp/weak$ ./a.out
no sqrt
```

在weak_main.c中定义的sqrt()触发已经生成的weak.so中的sqrt被动态链接到libc中的`sqrt`!