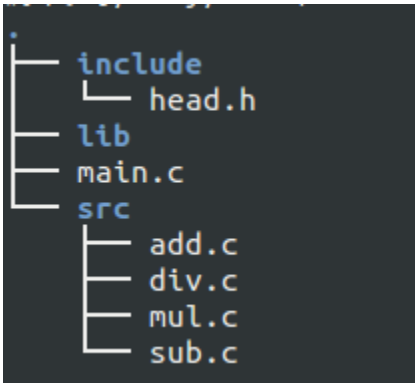
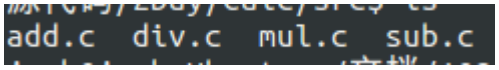
一.静态库的制作与使用

首先，有如下工作环境和目录：



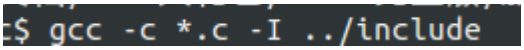
可以查看文件内容如下：

现在切换到 src 目录下，看到如下内容：

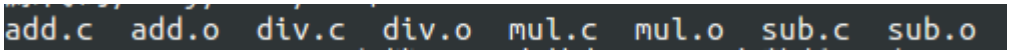


输入如下命令：

gcc -c *.c -I ../include

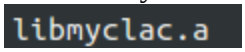


生成目标文件:

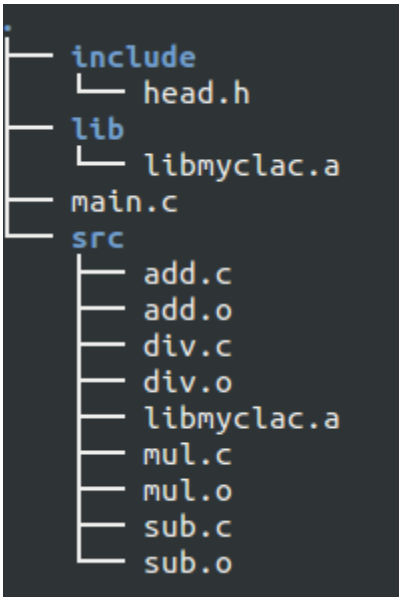


接下来生成静态库，输入如下命令：ar -rcs libmyclac.a *.o

获得库文件：



接下来将这个库文件拷贝到 lib 中，到时发布时就直接把 lib 文件夹和 include 文件夹发布给用户即可，调整后目录结构如下图所示：



现在编译测试下制作的库：

gcc main.c -I ./include/ -L ./lib/ -lmyclac -o app

```
$ gcc main.c -I ./include/ -L ./lib/ -lmyclac -o app
```

结果如下：

```
app  include  lib  main.c  src
```

测试如下：

```
源代码/2Day/Calc$ ./app
sum = 26
```

至此，静态库的制作与测试到此结束了。

二.动态库的制作与使用

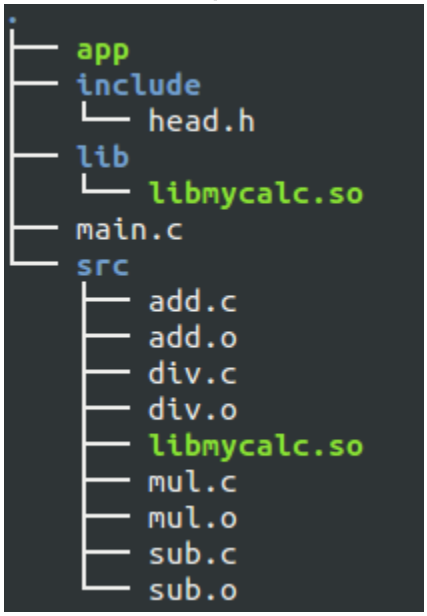
在上节 src 目录下敲入如下命令：gcc -shared *.o -o libmycalc.so

```
$ gcc -shared *.o -o libmycalc.so
```

结果入下：

```
add.c  add.o  div.c  div.o  libmycalc.so  mul.c  mul.o  sub.c  sub.o
```

接着，将 libmycalc.so 文件拷贝到 lib 目录下，文件夹结构如下图所示：



接下来编译测试程序：gcc main.c -I ./include/ -L ./lib/ -lmycalc -o app

```
$ gcc main.c -I ./include/ -L ./lib/ -lmycalc -o app
```

生成文件 app

```
app  include  lib  main.c  src
```

运行下，结果出错如下：

```
./app: error while loading shared libraries: libmycalc.so: cannot open shared object file: No such file or directory
```

首先查看文件类型: file app

```
源代码/2Day/Calc$ file app
app: ELF 64-bit LSB  executable, x86-64, version 1 (SYSV), dynamically linked (uses shared libs), for GNU/Linux 2.6.24, BuildID[sha1]=496ffdd2e3c6e095b
8525b9fffce1f5ffe724bc8, not stripped
```

可以看到是 elf 可执行文件，接着敲入如下命令:ldd app

```
源代码/2Day/Calc$ ldd app
        linux-vdso.so.1 =>  (0x00007ffdb1f95000)
        libmycalc.so => not found
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fbdfc922000)
        /lib64/ld-linux-x86-64.so.2 (0x000055dcda35c000)
```

可以看到，libmycalc.so 无法找到依赖。
然而，加载上面三个库需要/lib64/ld-linux-x86-64.so.2 (0x000055dcda35c000)加载。
也就是说需要找到依赖，要找到依赖就需要设置环境变量，查看环境变量:echo $PATH

```
源代码/2Day/Calc$ echo $PATH
/opt/FriendlyARM/toolchain/4.9.3/bin:/usr/lib/jvm/jdk1.6.0_45/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/ga
mes:/opt/FriendlyARM/toolchain/4.9.3/bin
```

可见，没有依赖库的环境路径。
所以需要设置环境变量，这里有两种方法设置。
第一种：

临时设置环境变量，输入如下命令：export LD_LIBRARY_PATH=./lib

```
$ export LD_LIBRARY_PATH=./lib
```

运行结果如下：

```
2Day/Calc$ ./app
sum = 26
```

查看相关依赖：

```
2Day/Calc$ ldd app
        linux-vdso.so.1 =>  (0x00007ffc5297a000)
        libmycalc.so => ./lib/libmycalc.so (0x00007f6c56504000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f6c5611d000)
        /lib64/ld-linux-x86-64.so.2 (0x000055d2712f9000)
```

方法没问题，问题是关闭终端后，环境失效。

第一种附加：

同样设置临时环境变量，输入如下命令：export LD_LIBRARY_PATH=./lib:$LD_LIBRARY_PATH

```
2Day/Calc$ export LD_LIBRARY_PATH=./lib:$LD_LIBRARY_PATH
```

运行结果如下：

```
2Day/Calc$ ./app
sum = 26
```

查看相关依赖库：

```
2Day/Calc$ ldd app
        linux-vdso.so.1 =>  (0x00007ffc9e248000)
        libmycalc.so => ./lib/libmycalc.so (0x00007f87036b2000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f87032cb000)
        /lib64/ld-linux-x86-64.so.2 (0x000055a740bfe000)
```

方法没有问题，但是关闭终端后环境无效！

第二种：

设置永久环境变量，有系统环境变量和用户环境变量之分。
下面举例说明：
1.设置系统环境变量：

a. 打开终端并输入：sudo gedit /etc/environment

b. 在 PATH="...."的末尾处添加：/opt/EmbedSky/4.3.3/bin

其中/opt/EmbedSky/4.3.3/bin 为你自己需要设置的环境变量路径

c. 使其立即生效，在终端执行：source /etc/environment

2.设置用户环境变量：

a. 打开终端并输入：sudo gedit ~/.bashrc

b. 前面的步骤会打开.bashrc 文件，在其末尾添加：

export PATH=/opt/EmbedSky/4.3.3/bin:$PATH

其中/opt/EmbedSky/4.3.3/bin 为你自己需要设置的环境变量路径

c. 使其立即生效，在终端执行：source ~/.bashrc

这样一来，问题可以得到比较好的解决！

当然，除了上面的修改环境变量方法以外还有一种方法，就是：修改/etc/ld.so.conf文件内容，将库的绝对路径添加到文件里面：

输入 ：sudo vi /etc/ld.so.conf

```
jack@jack-Ubuntu:~/Calc$ sudo vi /etc/ld.so.conf
```

修改文件如下：

```
jack@jack-Ubuntu: ~/Calc
1 include /etc/ld.so.conf.d/*.conf
2
3 /home/jack/Calc/lib/
```

关闭后退出，输入：sudo ldconfig -v

```
jack@jack-Ubuntu:~/Calc$ sudo ldconfig -v
```

结果看到输出一堆信息：

```
        libcomplzconfig.so.0 -> libcomplzconfig.so.0.0.0
        libgupnp-1.0.so.4 -> libgupnp-1.0.so.4.0.0
        libecal-1.2.so.16 -> libecal-1.2.so.16.0.0
        libisc.so.95 -> libisc.so.95.5.0
        libdmapsharing-3.0.so.2 -> libdmapsharing-3.0.so.2.9.24
        libvte2_90.so.9 -> libvte2_90.so.9.3400.9
        libmimic.so.0 -> libmimic.so.0.0.1
        libxapian.so.22 -> libxapian.so.22.6.3
        libpurple-client.so.0 -> libpurple-client.so.0.10.9
        libpeas-1.0.so.0 -> libpeas-1.0.so.0.801.0
        libsgutils2.so.2 -> libsgutils2.so.2.0.0
        libt1x.so.5 -> libt1x.so.5.1.2
        libebook-1.2.so.14 -> libebook-1.2.so.14.3.1
        libpspell.so.15 -> libpspell.so.15.2.0
        liblightdm-gobject-1.so.0 -> liblightdm-gobject-1.so.0.0.0
        libutempter.so.0 -> libutempter.so.1.1.5
jack@jack-Ubuntu:~/Calc$
```

查看链接信息: ldd app

```
jack@jack-Ubuntu:~/Calc$ ldd app
        linux-vdso.so.1 =>  (0x00007fffc01e3000)
        libmycalc.so => /home/jack/Calc/lib/libmycalc.so (0x00007f46a440a000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f46a4041000)
        /lib64/ld-linux-x86-64.so.2 (0x0000561e634da000)
```

运行程序：

```
jack@jack-Ubuntu:~/Calc$ ./app
sum = 26
```

同样可以达到效果！