**构建主机端开发环境**

**组名：The three stooges**

**小组成员：曾毓宁、徐锦畅、唐培津**

**一、实验目的**

**1、在建立交叉编译开发平台之前，首先需要建立主**

**机(虚拟机/PC，或双系统）开发环境； **

**2、学会使用本地gcc编译应用程序； **

**3、学会使用Makefile管理应用程序； **

**4、学会通过autotools生成Makefile，学会常用的**

**make操作； **

**5、学会通过git/github管理团队软件和工作文件。**

**二、实验内容**

**在基于linux系统的虚拟机上编写C程序，完成gcc手动编译、手动生成makefile的make编译、autotools自动生成makefile的make编译，并学会常用的make操作。最后使用github来管理小组的文件。**

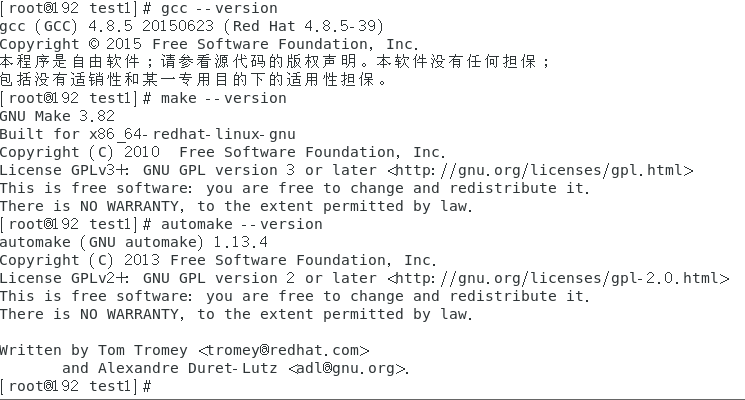
1. **实验过程与结果**

**（1）centos**

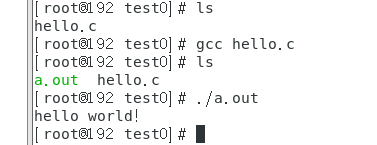
1. **在VMware workstation 上建立了虚拟机开发环境，我的开发环境是centos 7。因为对linux不太熟练，所以安装时选择GUI，便于学习理解；**



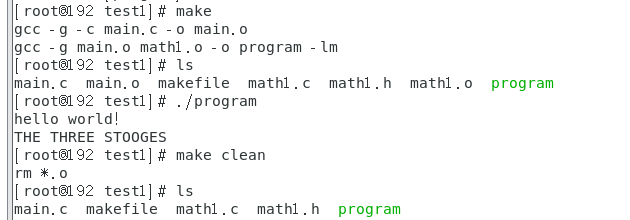
1. **安装gcc、make以及autotools。这里直接用指令：yum -y install gcc gcc-c++ kernel-devel即可安装gcc、c++编译器以及内核文件。安装结束后用指令gcc --version即可查看版本信息，确定安装成功。Make和autotools也可用上述方法查看。**

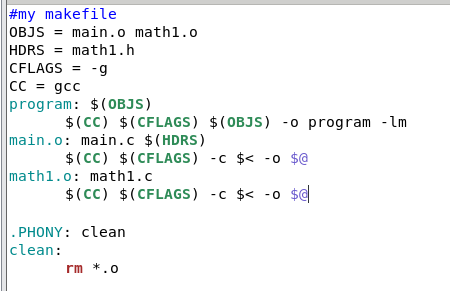


1. **编写一个简单的C程序。这里我写了一个输出hello world的程序。然后用gcc进行编译，生成一个a.out文件。使用指令 ./a.out运行，结果正确，成功使用gcc进行编译。**



1. **编写几个程序组成小工程。这里我写了三个程序：main.c、math1.c以及math1.h。Math1.c文件中的函数输出THE THREE STOOGES，math1.h是头文件，main.c中输出hello world并调用math1.c中的函数。然后编写makefile文件，实现自动编译。**

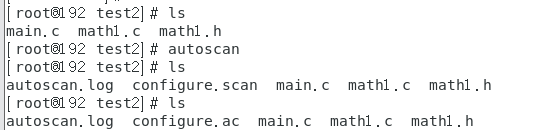




**在手动编写makefile的过程中，我遇到了一个问题，执行make报错：Makefile:6:\*\*\*遗漏分隔符。停止 。经过查询相关资料发现，gcc、rm等命令前面是tab而不是空格，改成tab后可以正常执行。**

1. **还是用上面的程序，这里使用autotools生成makefile。**

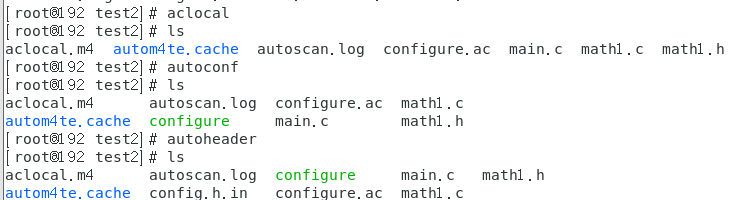
**首先运行autoscan命令扫描test2目录，生成configure.scan文件。Configure.scan文件其实就是模板，我们将该文件进行手动修改后将其重命名为configure.ac文件。**



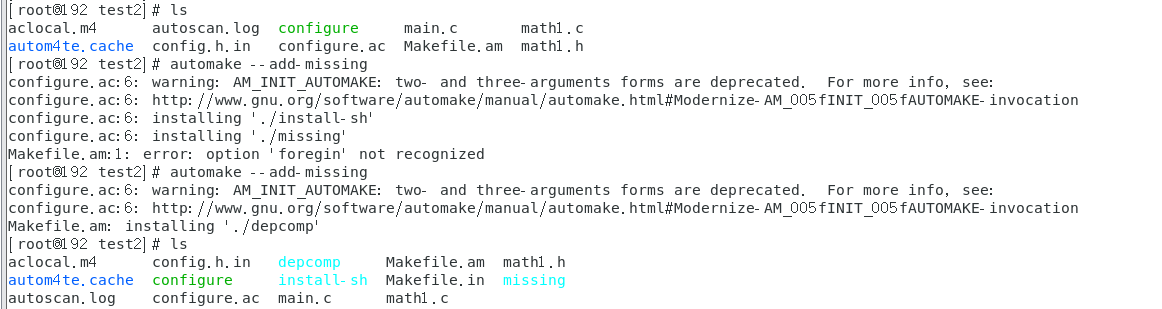
**接着，使用aclocal扫描该文件，生成aclocal.m4，获取autocinf创建configure所需的宏定义。**

**再使用autoconf读取configure.ac中的宏，生成configure脚本。**

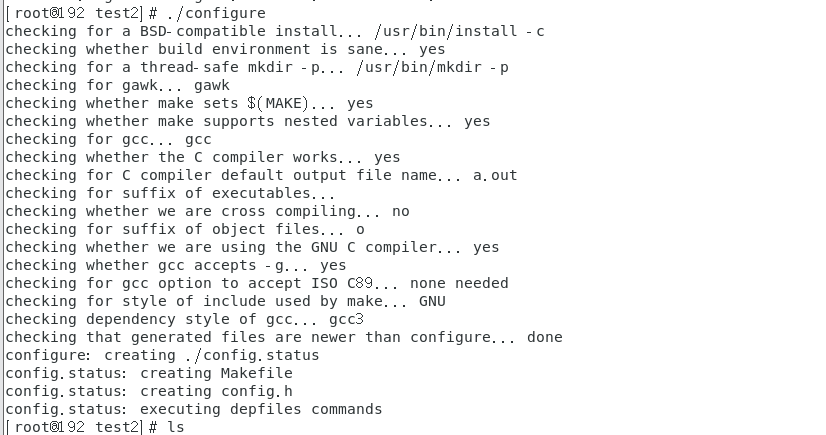
**接着使用autoheader命令生成config.h.in文件。**

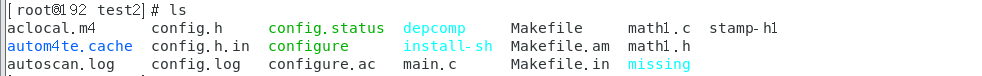


**随后手动创建Makefile.am文件。Automake根据configure.ac的参数把Makefile.am转换成Makefile.in文件。**

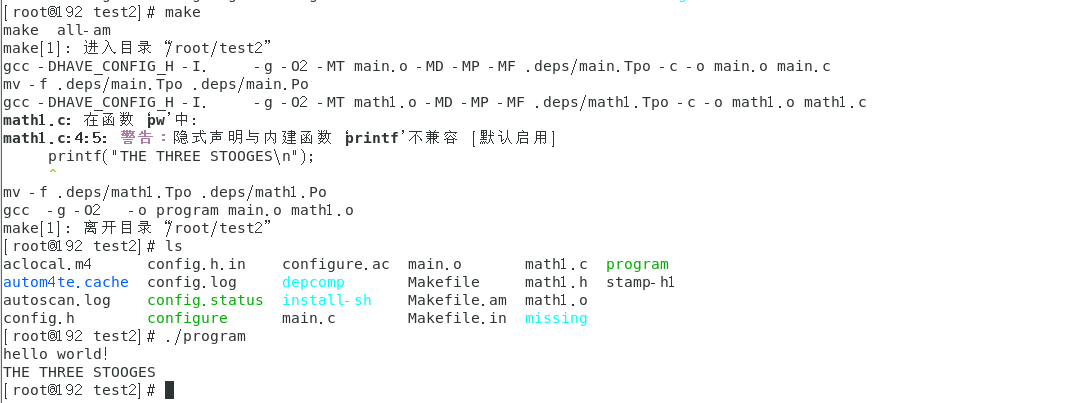


**使用./configure生成最终的Makefile文件。**



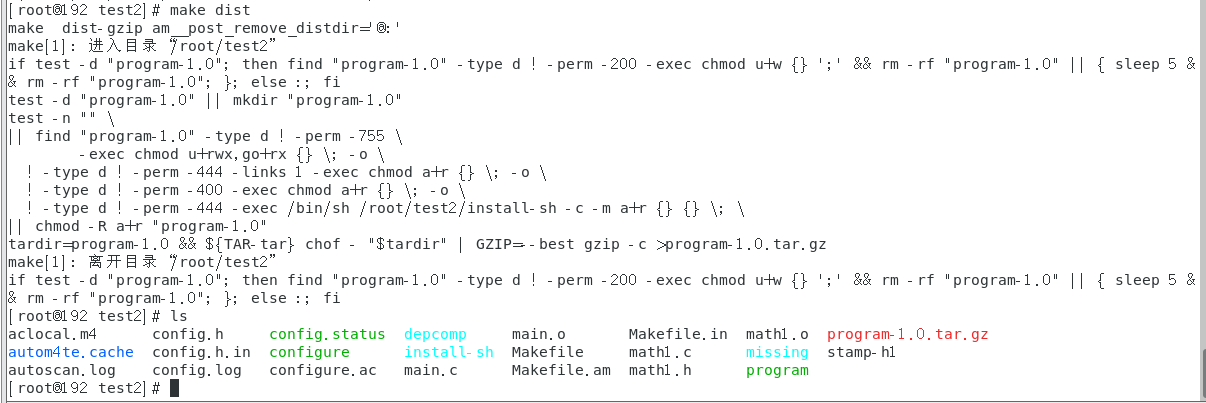


**运行make测试文件生成是否正确。**



**最后我尝试了make install、make unstall以及make dist命令，均能成功运行。**





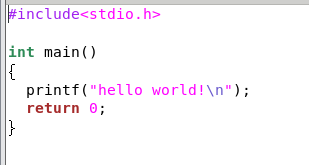
**四、实验总结**

**这是本学期第一次实验，我们利用本次实验熟悉了linux的一些常用指令，同时学会使用了gcc编译器、make以及automake等工具，感觉收获颇丰。之前写程序基本都是在win的环境下，可能对linux不太熟悉。在后续学习中，我们会勤加练习，争取早日熟练掌握linux下的各种操作。**

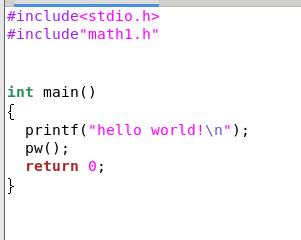
**五、附实验源码**

**（1）Centos：**

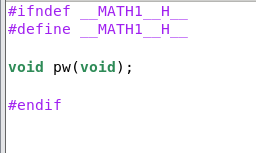
**Hello.c:**



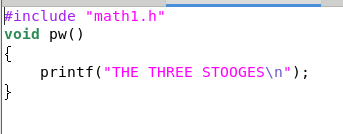
**Main.c:**



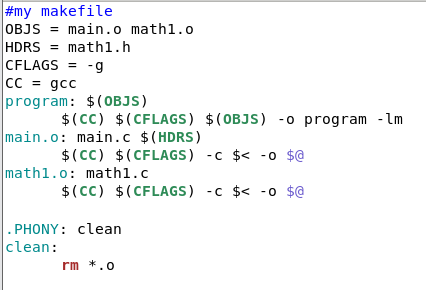
**Math1.h:**



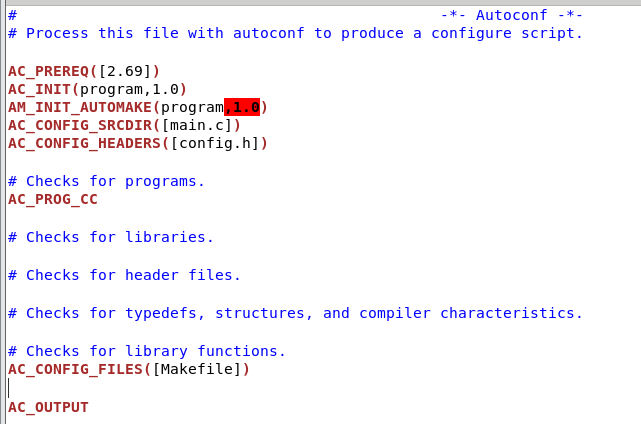
**Math1.c:**



**手动编写的makefile：**



**Configure.ac:**



**手动编写的Makefile.am:**

