

Virtual Machine Linux Report

计算机238班 李政庭

Experimental requirement

1. 在你的虚拟机上安装 CentOS 系统，并配置好环境。
2. 使用**命令行**编写一个简单的 C++ 程序，该程序实现以下功能：
 - 输出 "Hello, World!"
 - 计算并输出一个简单的数学表达式，例如 $f(x) = x^2 + 2x + 1$ ，当 $x = 5$ 时的结果
3. 编译并运行你的 C++ 程序。
4. 完成报告的编写，并包含必要的命令行操作截图。
5. 如果可以，请将项目传至 GitHub（本项可以不完成）。

Experimental Step

1.Install CentOS

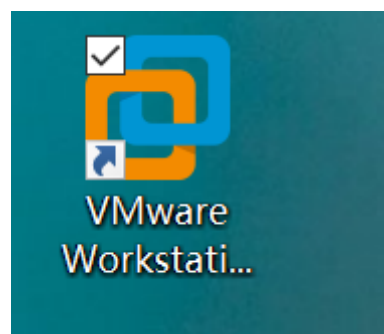
Firstly, I installed the CentOS 7 from [[Index of /centos-vault/7.6.1810/isos/x86_64/](#) | [清华大学开源软件镜像站](#) | [Tsinghua Open Source Mirror](#)]. Then i chose `CentOs-7-x86_64-DVD-1810.iso` to install.

`CentOS-7-x86_64-DVD-1810.iso`

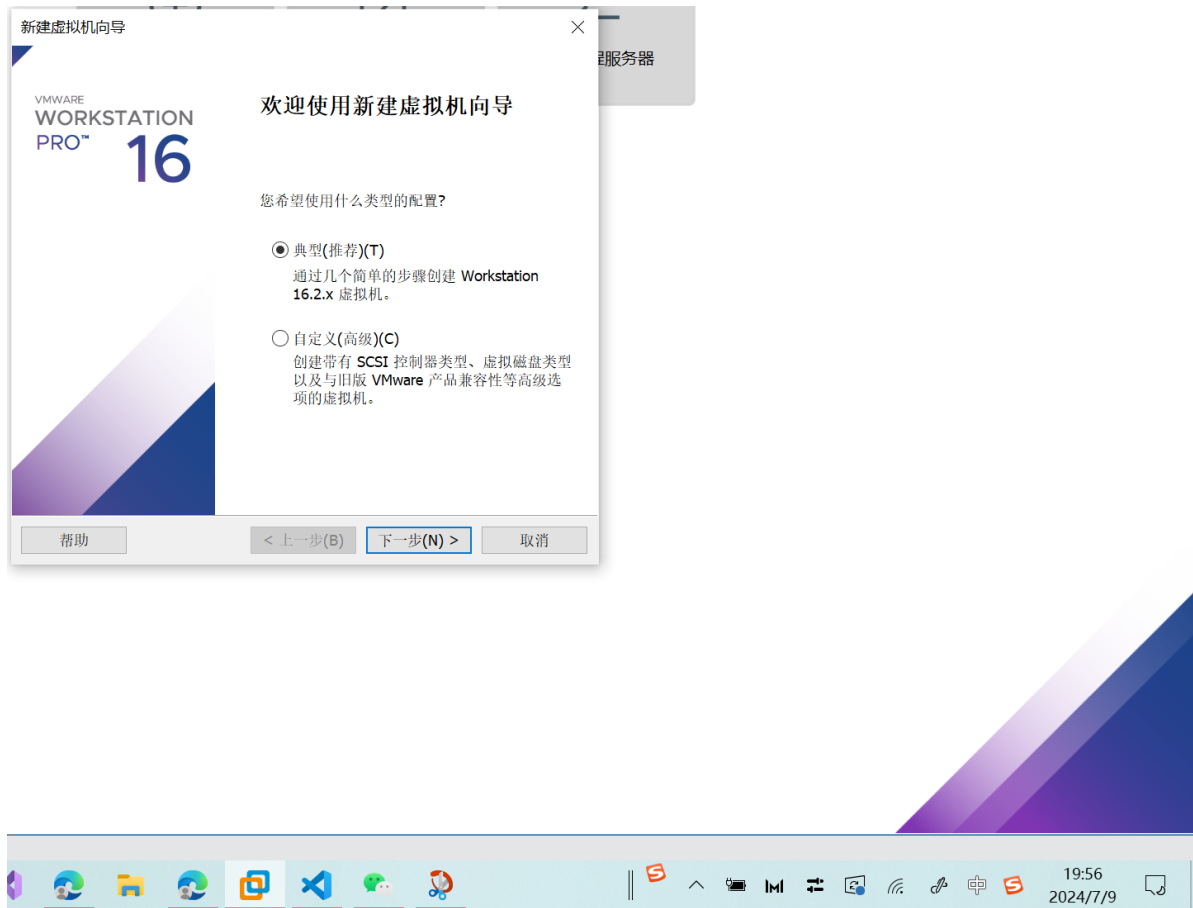
4.3 GiB

2018-11-26 07:55

Secondly, I installed VMware WorkStation Pro 16 from[[下载 VMware Workstation Pro | CN](#)].



Then I started to build a virtual machine and configured the environment. The key steps as the follow pictures:



安装客户机操作系统

虚拟机如同物理机，需要操作系统。您将如何安装客户机操作系统？

安装来源：

☐ 安装程序光盘(D):

无可用驱动器

☐ 安装程序光盘映像文件(iso)(M):

D:\edg浏览器下载\CentOS-7-x86_64-DVD-1810.iso

浏览(R)...

☒ 稍后安装操作系统(S)。

创建的虚拟机将包含一个空白硬盘。

帮助

< 上一步(B)

下一步(N) >

取消

选择客户机操作系统

此虚拟机中将安装哪种操作系统？

客户机操作系统

- ☐ Microsoft Windows(W)
- ☒ Linux(L)
- ☐ VMware ESX(X)
- ☐ 其他(O)

版本(V)

Red Hat Enterprise Linux 7 64 位



帮助

< 上一步(B)

下一步(N) >

取消

简易安装信息

这用于安装 CentOS 7 64 位。

个性化 Linux

全名(F):

Test

用户名(U):


test02

密码(P):

●●●●●●

确认(C):

●●●●●●

 用户帐户和根帐户均使用此密码。

帮助

< 上一步(B)

下一步(N) >

取消

命名虚拟机

您希望该虚拟机使用什么名称？

虚拟机名称(V):

位置(L):

在“编辑”>“首选项”中可更改默认位置。

指定磁盘容量

磁盘大小为多少？

虚拟机的硬盘作为一个或多个文件存储在主机的物理磁盘中。这些文件最初很小，随着您向虚拟机中添加应用程序、文件和数据而逐渐变大。

最大磁盘大小 (GB)(S):

针对 CentOS 7 64 位 的建议大小: 20 GB

- ☐ 将虚拟磁盘存储为单个文件(O)
- ☒ 将虚拟磁盘拆分成多个文件(M)

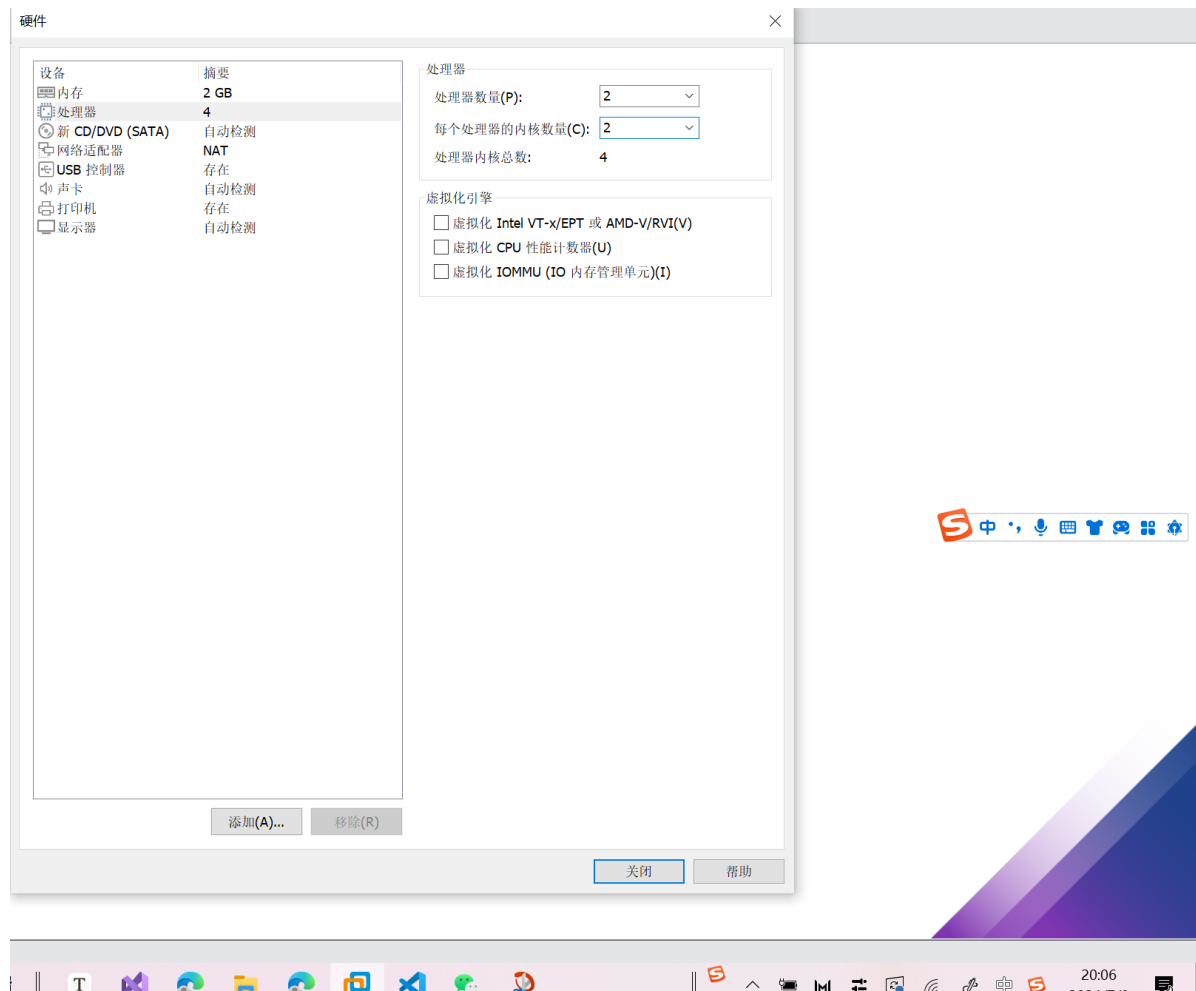
拆分磁盘后，可以更轻松地在计算机之间移动虚拟机，但可能会降低大容量磁盘的性能。

帮助

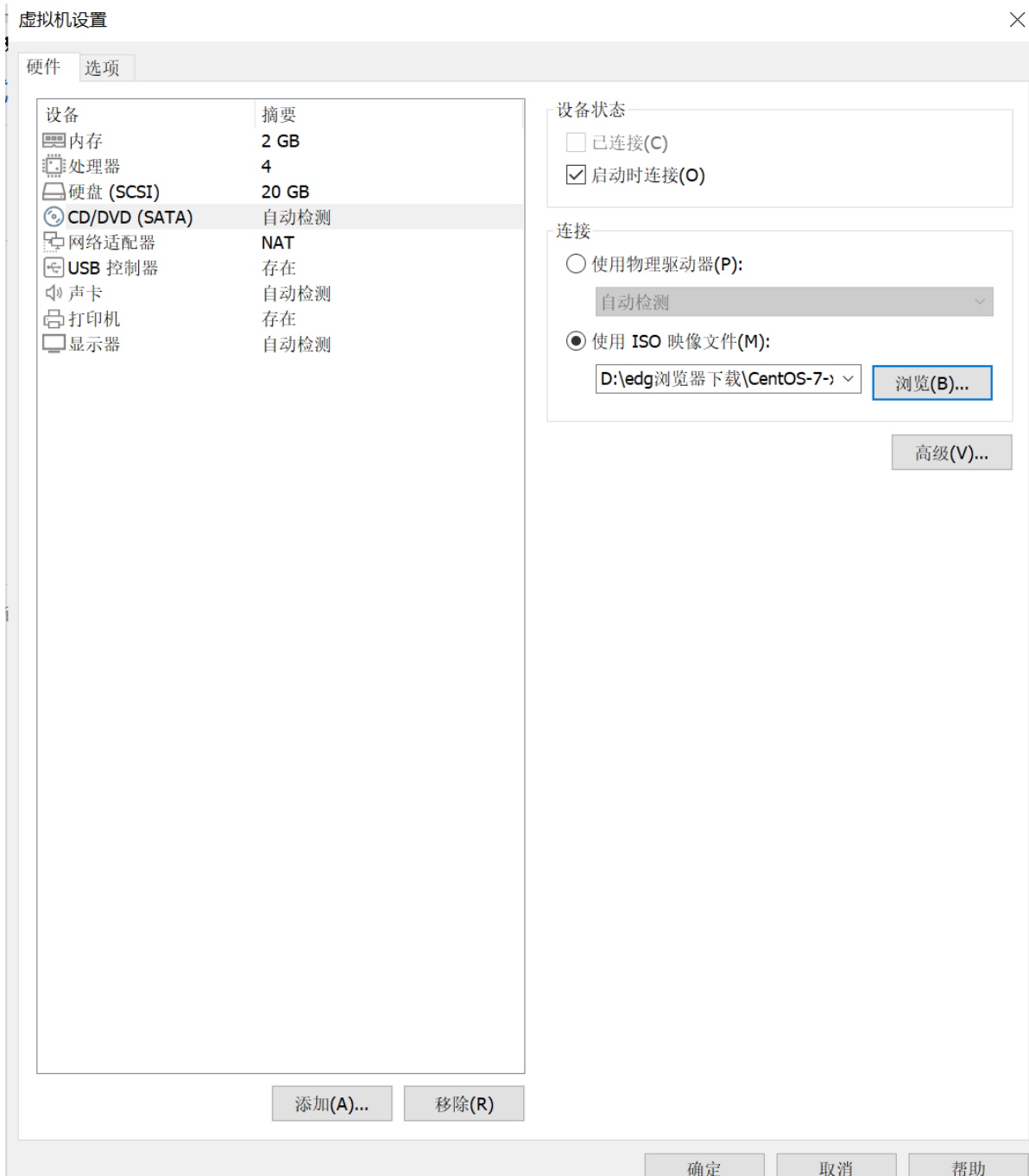
< 上一步(B)

下一步(N) >

取消



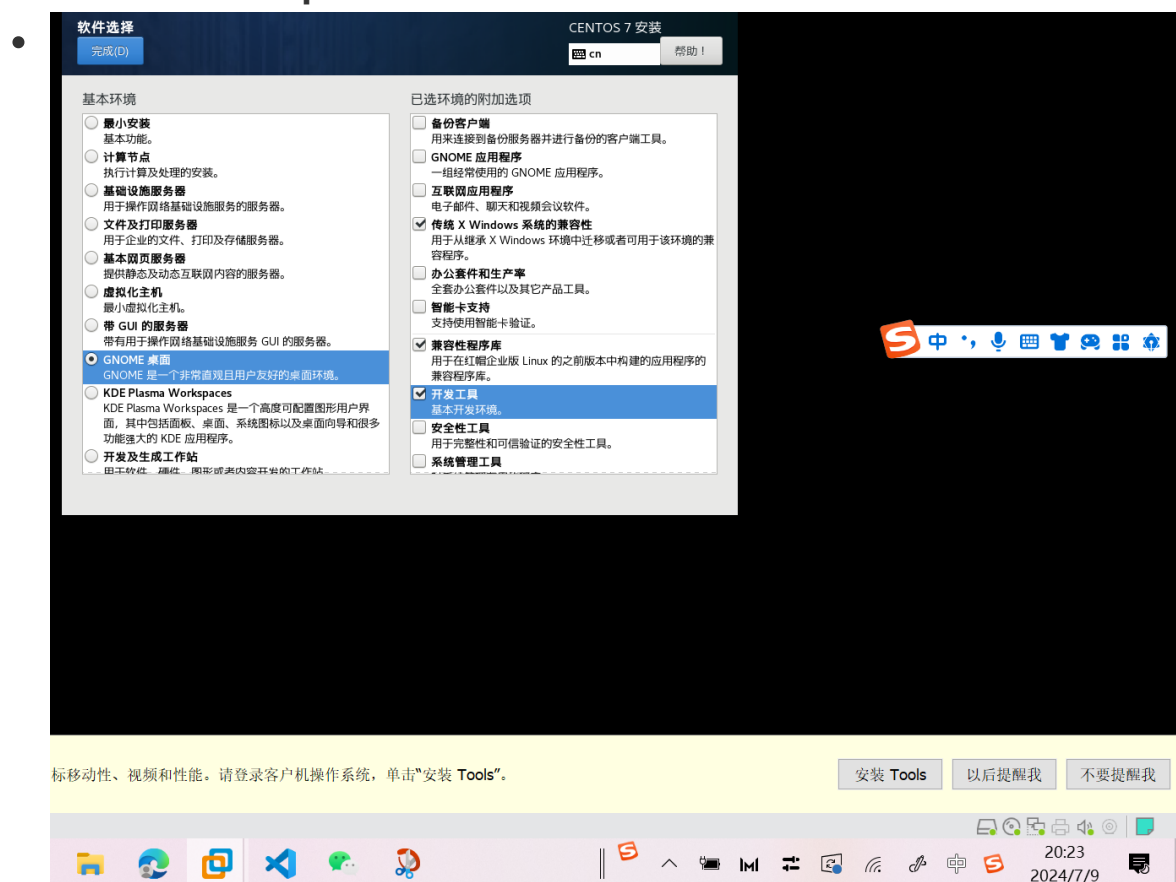
- In this part, I should select the file i downloaded from `CentOS-7-x86_64-DVD-1810.ios` as the IOS image file.



8. Next, start my virtual machine.



- 9. In this part , click the software choose . The choice is as the follow picture:



- The development tools contains gcc

10. configure the download position



安装目标位置

CENTOS 7 安装

完成(D)

cn

帮助 !

设备选择

选择要在其中安装系统的设备。点击主菜单中的“开始安装”按钮前不会对该设备进行任何操作。

本地标准磁盘

20 GiB

VMware, VMware Virtual S

sda / 992.5 KiB 空闲

不会对未在此处选择的磁盘进行任何操作。

专用磁盘 & 网络磁盘

添加磁盘(A)...

不会对未在此处选择的磁盘进行任何操作。

其它存储选项

分区

☐ 自动配置分区(U)。

☒ 我要配置分区(I)。

☐ 我想让额外空间可用(M)。

加密

完整磁盘摘要以及引导程序(F)...

已选择 1 个磁盘 ; 容量 20 GiB ; 992.5 KiB 空闲 [刷新\(R\)](#)

手动分区

CENTOS 7 安装

完成(D)

cn

帮助 !

▼ 新 CentOS 7 安装

系统

/boot

sda1

1023 MiB

>

/

sda2

17 GiB

swap

sda3

2048 MiB

+

-

↺

可用空间

992.5 KiB

总空间

20 GiB

[已选择 1 个存储设备\(S\)](#)

[全部重设\(R\)](#)

sda1

挂载点(P) :

/boot

期望容量(D) :

1023 MiB

设备类型(T) :

标准分区

☐ 加密(E)

文件系统(Y) :

ext4

☒ 重新格式化(O)

设备 :

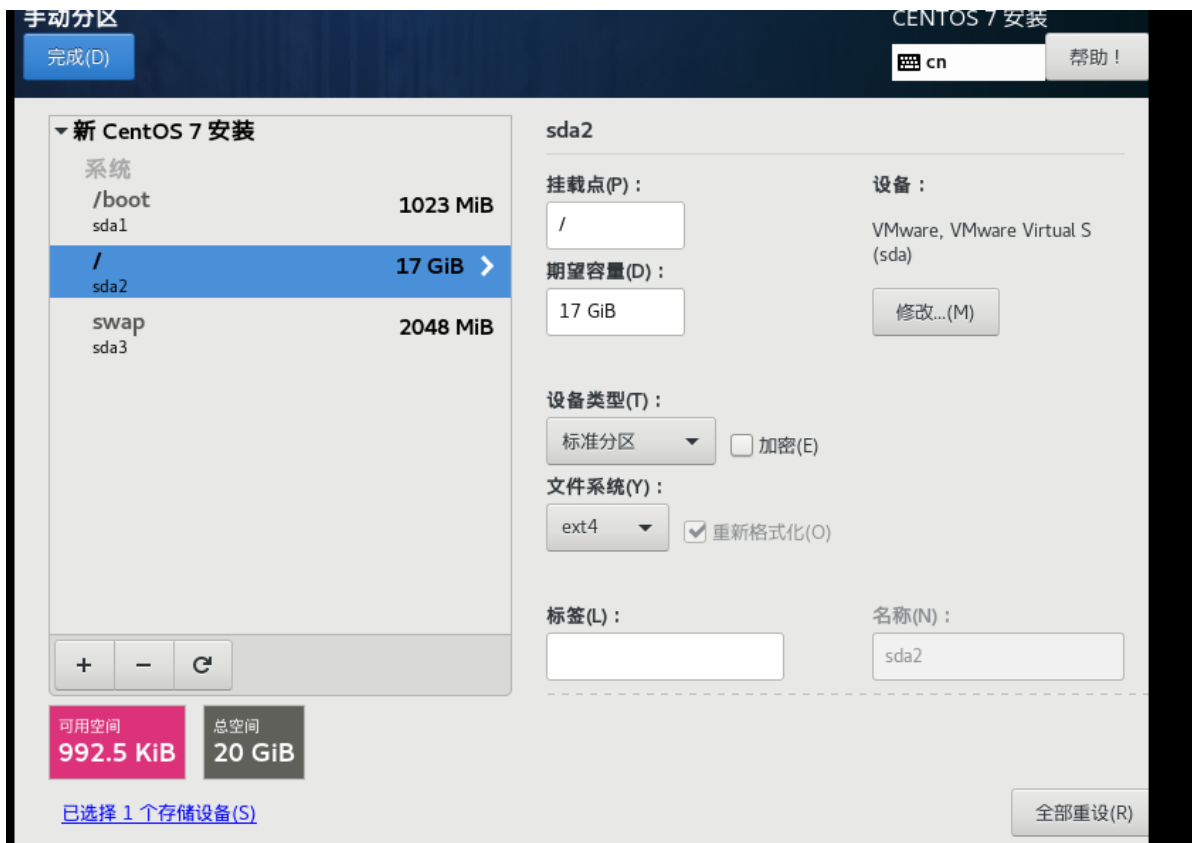
VMware, VMware Virtual S (sda)

修改...(M)

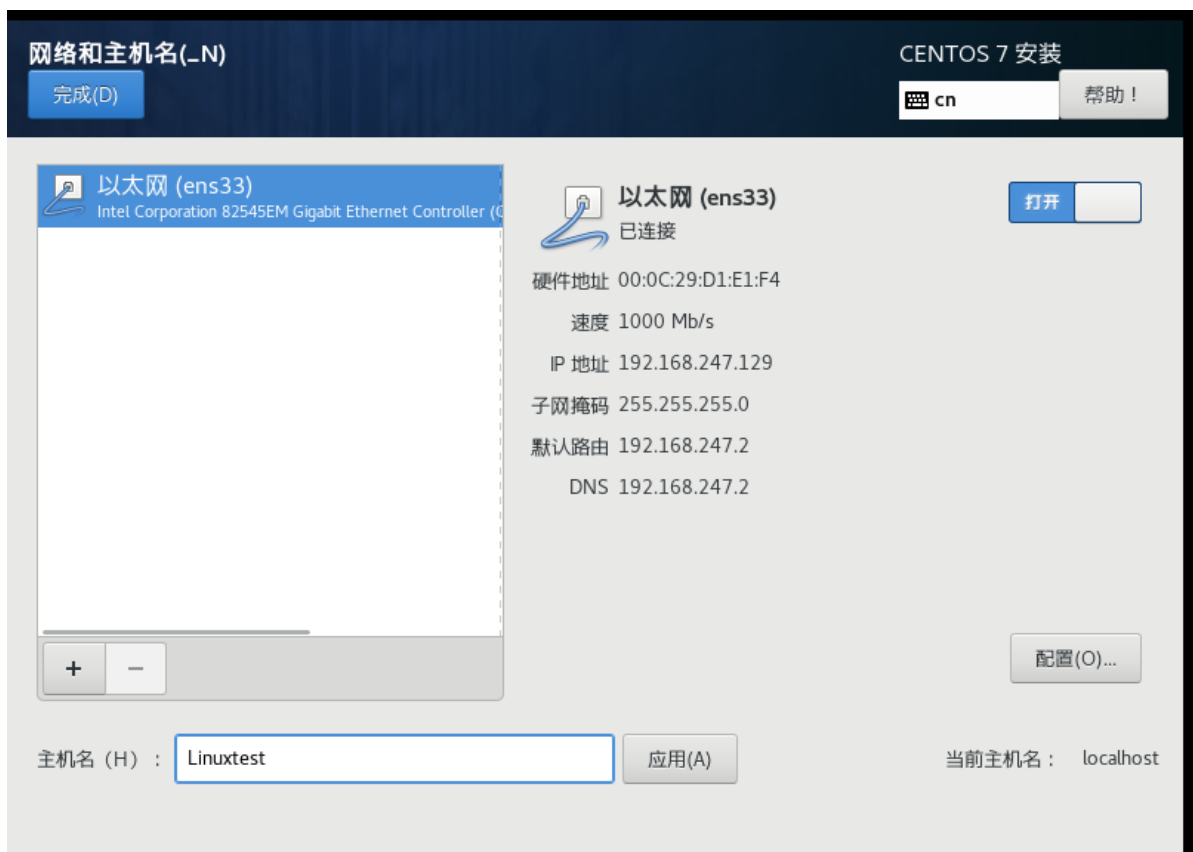
标签(L) :

名称(N) :

sda1

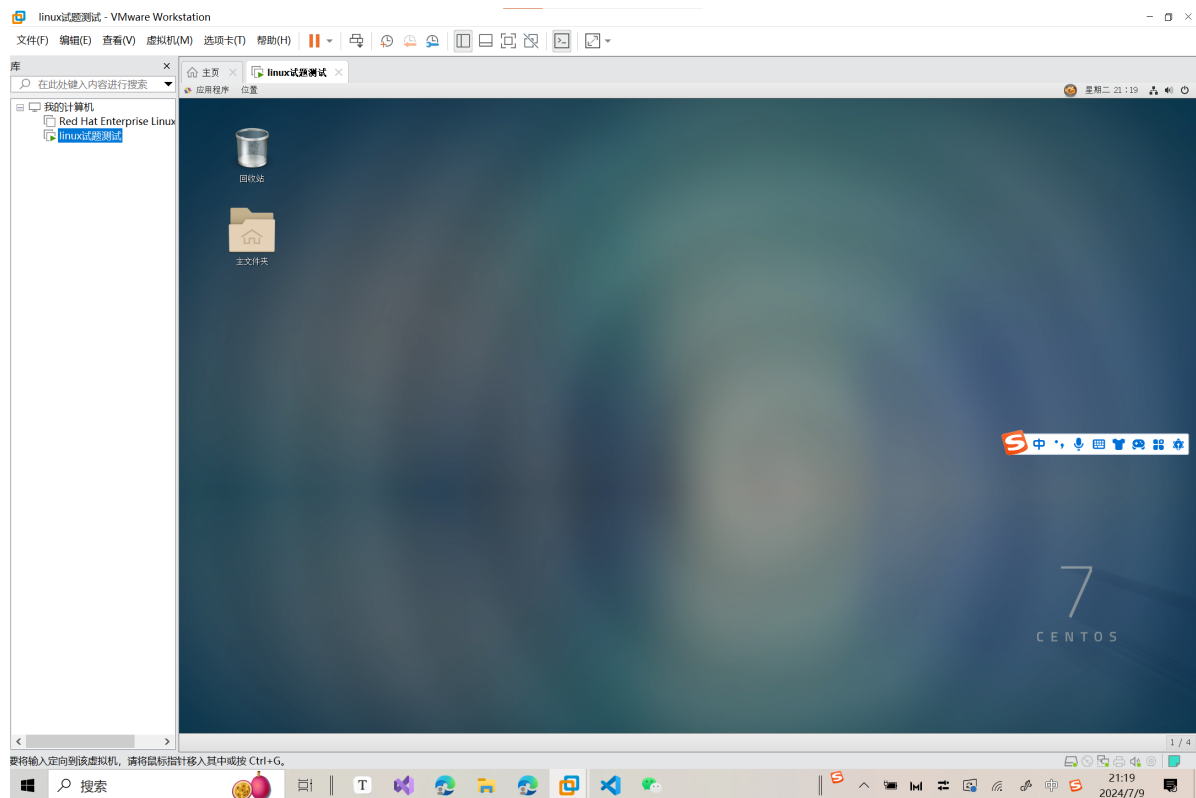


11. Set the Internet



Difficulties that I met: This is my first time to learn some knowledge about virtual machine and Linux Operate System. So i don't have any experience about these things. To solve this problem, I read some blogs from CSDN and watched some related videos about Linux. After I have a holistic concept about Linux and the history of Linux, I started to download virtual machine software and CentOS file. Then I configured the basic parameters follow the blogs. In a nutshell, the whole process of exploration is an enhancement of my ability of retrieving information.

configure compile environment



- Because I already have chose the development tools before, so the system already have g++ and gcc.

```
root@Linuxtest:~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
[root@Linuxtest ~] # gcc -v  
使用内建 specs。  
COLLECT_GCC=gcc  
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/4.8.5/lto-wrapper  
目标: x86_64-redhat-linux  
配置为: ../configure --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-bootstrap --enable-shared --enable-threads=posix --enable-checking=release --with-system-zlib --enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-linker-hash-style=gnu --enable-languages=c,c++,obj-c,obj-c++,java,fortran,ada,go,lto --enable-plugin --enable-initfini-array --disable-libgcj --with-isl=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86_64-redhat-linux/isl-install --with-cloog=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86_64-redhat-linux/cloog-install --enable-gnu-indirect-function --with-tune=generic --with-arch_32=x86-64 --build=x86_64-redhat-linux  
线程模型: posix  
gcc 版本 4.8.5 20150623 (Red Hat 4.8.5-36) (GCC)  
[root@Linuxtest ~] # s
```

```

[root@Linuxtest ~]# g++ -v
使用内建 specs。
COLLECT_GCC=g++
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/4.8.5/lto-wrapper
目标：x86_64-redhat-linux
配置为：../configure --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/
info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-bootstrap --enab
le-shared --enable-threads=posix --enable-checking=release --with-system-zlib --
enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-object --
enable-linker-build-id --with-linker-hash-style=gnu --enable-languages=c,c++,obj
c,obj-c++,java,fortran,ada,go,lto --enable-plugin --enable-initfini-array --disa
ble-libgcj --with-isl=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86_64-redhat
-linux/isl-install --with-cloog=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86
_64-redhat-linux/cloog-install --enable-gnu-indirect-function --with-tune=generi
c --with-arch_32=x86-64 --build=x86_64-redhat-linux
线程模型：posix
gcc 版本 4.8.5 20150623 (Red Hat 4.8.5-36) (GCC)
[root@Linuxtest ~]# s

```

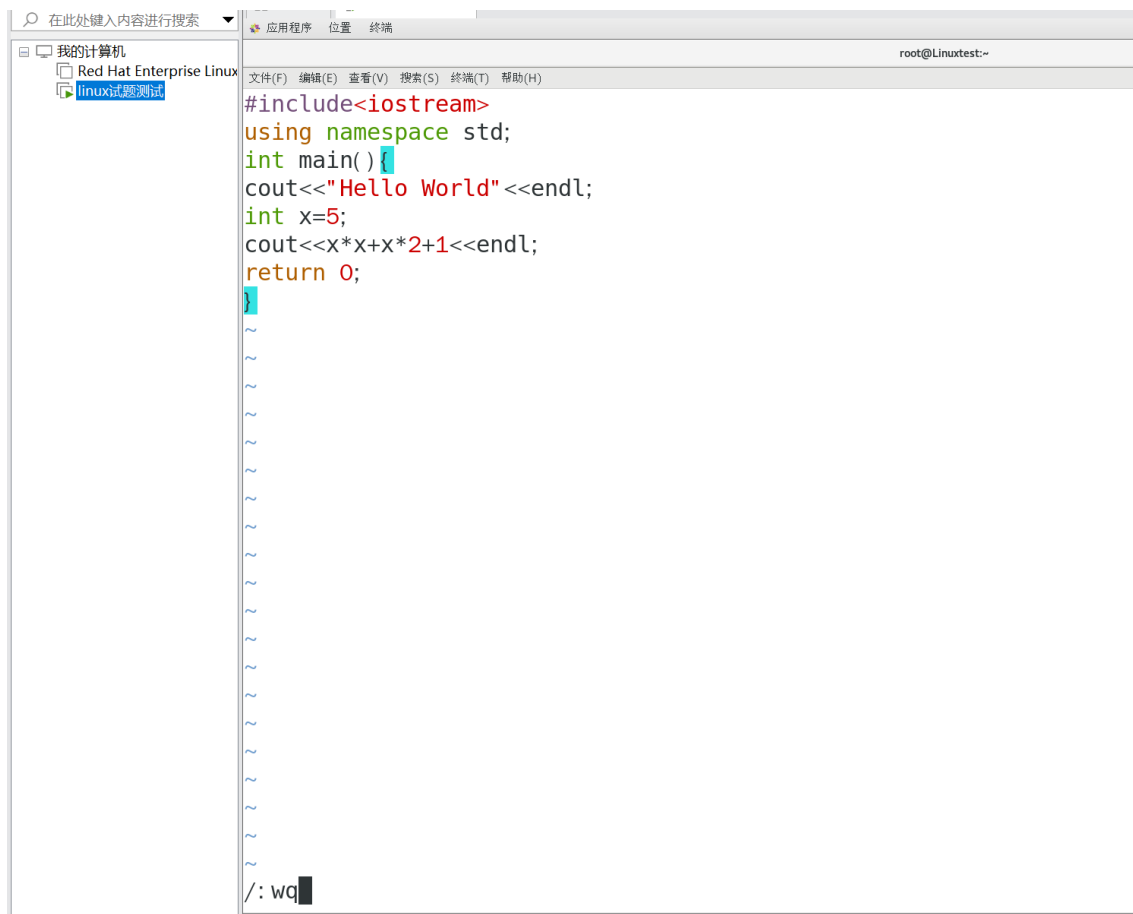
So if my CentOS didn't configure GCC before, it can be solved by these steps:

Open the terminal and input `sudo yum update` and `sudo yum install gcc` and `sudo yum install gcc-c++`

Difficulties that I met: The CentOS is totally different from Windows which i familiar with. Most of operations are on the terminal command line, unlike Windows. So I was confused at first , and made a lot of mistake when I was typing commands. But fortunately I successfully finished configuring the CentOS in following the information I had gathered.

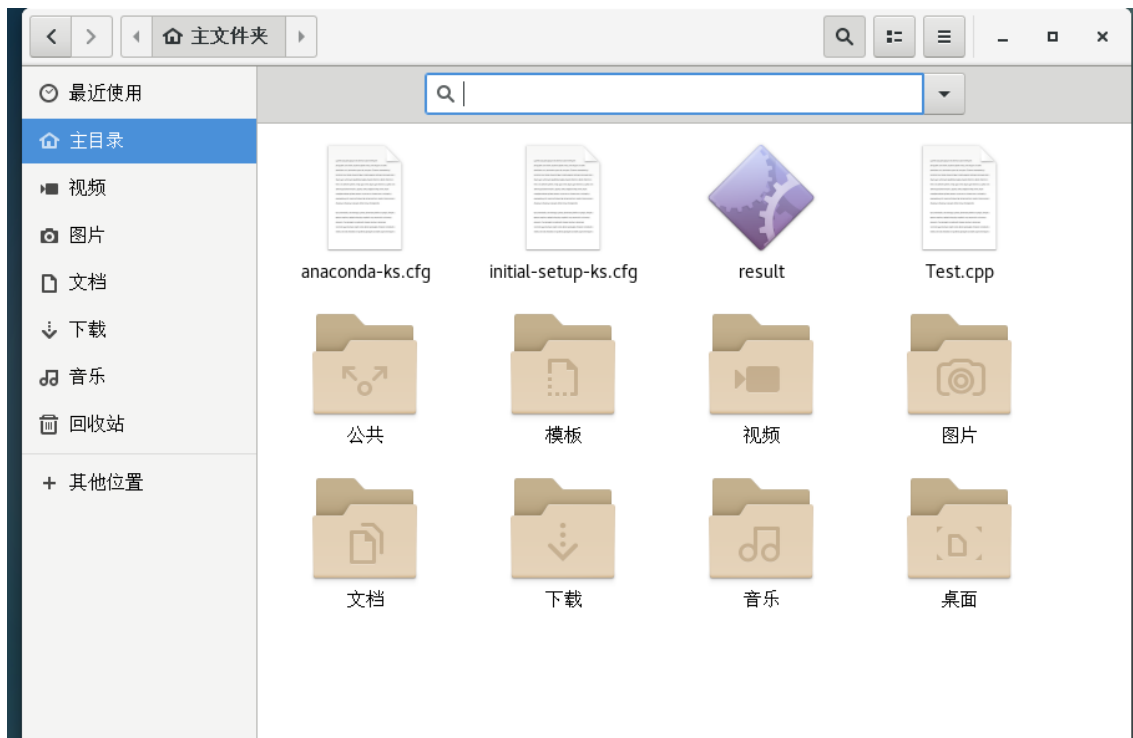
Use the command line to write C++ program

- Open the terminal and input `vim Test.cpp` then i can start to code.



```
#include<iostream>
using namespace std;
int main(){
cout<<"Hello World"<<endl;
int x=5;
cout<<x*x*x*2+1<<endl;
return 0;
}
```

The file is indeed exist.



- input `g++ -o name filename.cpp` to compile the file and input `./name` to run my program.


```
[root@Linuxtest ~]# vim Test.cpp
[root@Linuxtest ~]# vi
[root@Linuxtest ~]# g++ -o result Test.cpp
[root@Linuxtest ~]# ./result
bash: ./result: 没有那个文件或目录
[root@Linuxtest ~]# ./result
Hello World
36
```

Difficulties that I met: Although i know how to make a C++ program, it is my first time to use vim. So i have no idea how to use it. So i read some blogs about vim. knowing that vim has three modes: Command Mode, Insert Mode and Last Line Mode. Actually, it is not difficult to handle it after few practice. Therefore, this part is relative easy to accomplish.

Send to github

I have already put the PDF to my github repository <https://github.com/Leezhengting/NCUSCC-.git>

The screenshot shows the GitHub interface for a repository named 'NCUSCC-' owned by 'Leezhengting'. The repository is public and has 1 branch (main) and 0 tags. The file list shows 'README.md' as the initial commit, made 2 days ago. There is also a file named '李政庭9109223234Virtual Machine Linux Repo...' which was added via upload 'now'. The repository has 8 commits in total.

File	Commit Message	Commit Date
README.md	Initial commit	2 days ago
李政庭9109223234Virtual Machine Linux Repo...	Add files via upload	now