实验一

实验报告

计算机科学与技术系

191220008 陈南曈

一、实验目的

1. 学会自己安装 Linux 系统

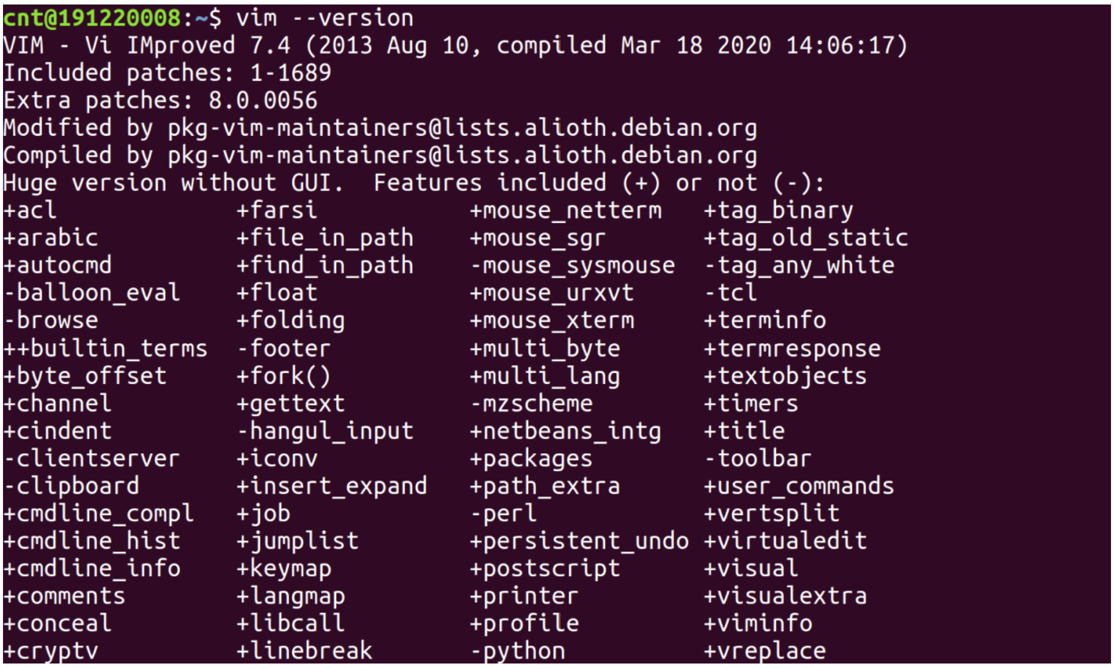
2. 学会配置简单的 Linux 开发环境

3. 在 Linux 下完成简单编程练习并熟悉各种命令行工具的使用方法

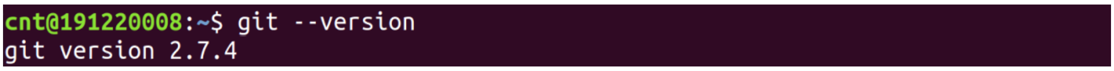
二、实验内容

1、Linux 的安装和配置

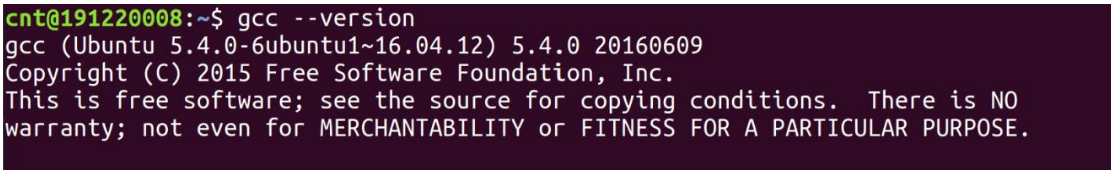
ii. 使用 man 查询 vim/git/gcc/as/objdump/gdb 版本的命令，然后使用查到的命令打印出对应版本



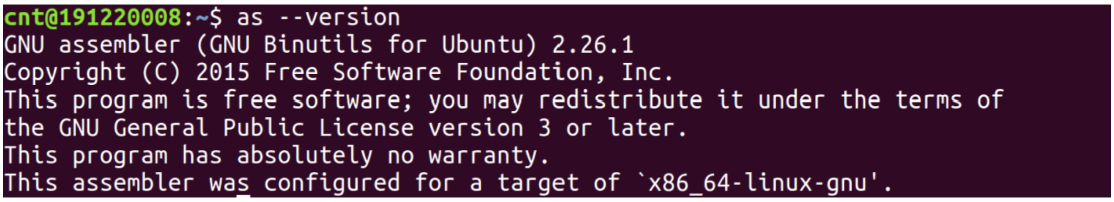
```
cnt@191220008:~$ vim --version
VIM - Vi IMproved 7.4 (2013 Aug 10, compiled Mar 18 2020 14:06:17)
Included patches: 1-1689
Extra patches: 8.0.0056
Modified by pkg-vim-maintainers@lists.alioth.debian.org
Compiled by pkg-vim-maintainers@lists.alioth.debian.org
Huge version without GUI.  Features included (+) or not (-):
+acl             +farsi           +mouse_netterm   +tag_binary
+arabic          +file_in_path    +mouse_sgr       +tag_old_static
+autocmd         +find_in_path    -mouse_sysmouse  -tag_any_white
-balloon_eval    +float           +mouse_urxvt     -tcl
-browse          +folding         +mouse_xterm     +terminfo
++builtin_terms  -footer          +multi_byte      +termresponse
+byte_offset     +fork()          +multi_lang      +textobjects
+channel         +gettext         -mzscheme        +timers
+cindent         -hangul_input    +netbeans_intg   +title
-clientserver    +iconv           +packages        -toolbar
-clipboard       +insert_expand   +path_extra      +user_commands
+cmdline_compl   +job             -perl            +vertsplit
+cmdline_hist    +jumplist        +persistent_undo +virtualedit
+cmdline_info    +keymap          +postscript      +visual
+comments        +langmap         +printer         +visualextra
+conceal         +libcall         +profile         +viminfo
+cryptv          +linebreak       -python          +vreplace
```



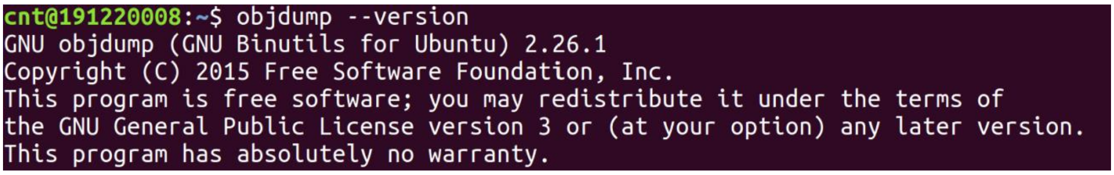
```
cnt@191220008:~$ git --version
git version 2.7.4
```



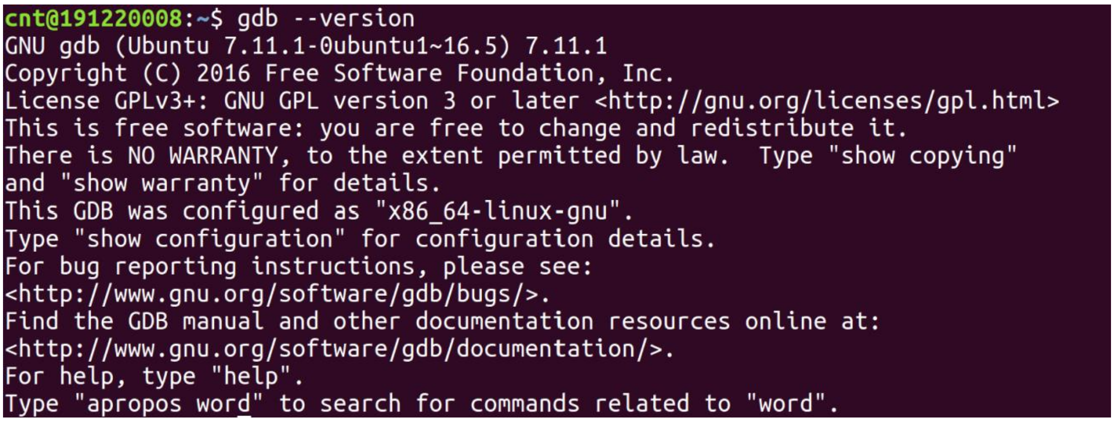
```
cnt@191220008:~$ gcc --version
gcc (Ubuntu 5.4.0-6ubuntu1~16.04.12) 5.4.0 20160609
Copyright (C) 2015 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```



```
cnt@191220008:~$ as --version
GNU assembler (GNU Binutils for Ubuntu) 2.26.1
Copyright (C) 2015 Free Software Foundation, Inc.
This program is free software; you may redistribute it under the terms of
the GNU General Public License version 3 or later.
This program has absolutely no warranty.
This assembler was configured for a target of `x86_64-linux-gnu'.
```



```
cnt@191220008:~$ objdump --version
GNU objdump (GNU Binutils for Ubuntu) 2.26.1
Copyright (C) 2015 Free Software Foundation, Inc.
This program is free software; you may redistribute it under the terms of
the GNU General Public License version 3 or (at your option) any later version.
This program has absolutely no warranty.
```



```
cnt@191220008:~$ gdb --version
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
```

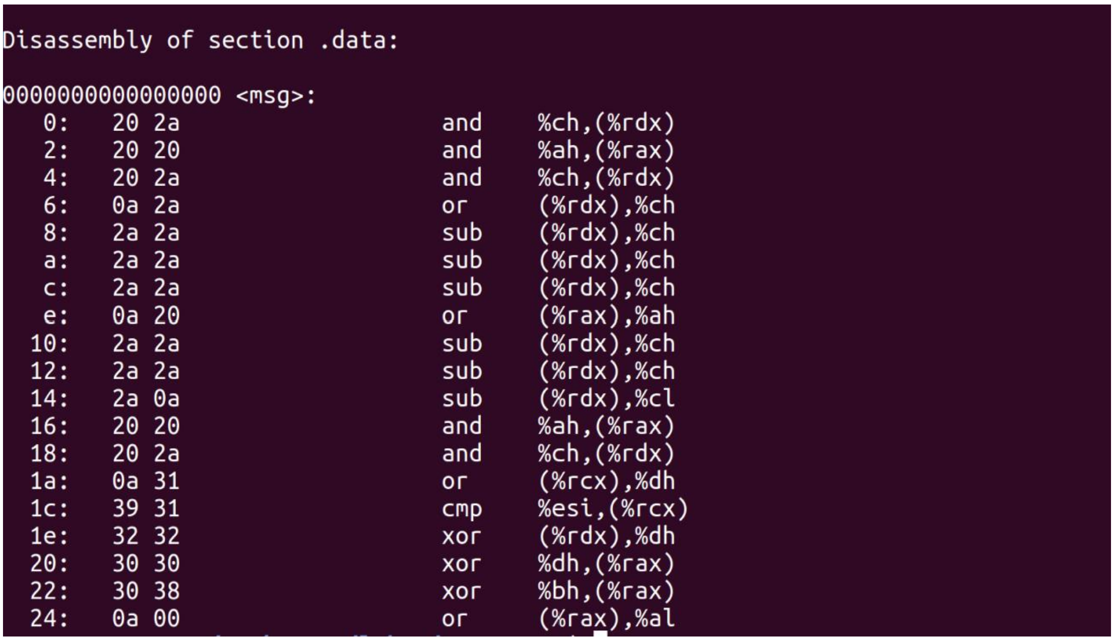
（gdb 的版本查看方式未在 man 中找到，猜测可以用--version 查看，事实也的确如此）

iii. 写下你在安装过程中遇到的问题，并说明你是如何解决的

（无）之前已经安装完毕

3、熟悉工具

i. 使用 objdump 的-D 选项反汇编 heart.o 文件，找到你学号的位置

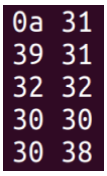


```
Disassembly of section .data:

0000000000000000 <msg>:
   0:   20 2a                   and    %ch,(%rdx)
   2:   20 20                   and    %ah,(%rax)
   4:   20 2a                   and    %ch,(%rdx)
   6:   0a 2a                   or     (%rdx),%ch
   8:   2a 2a                   sub    (%rdx),%ch
   a:   2a 2a                   sub    (%rdx),%ch
   c:   2a 2a                   sub    (%rdx),%ch
   e:   0a 20                   or     (%rax),%ah
  10:   2a 2a                   sub    (%rdx),%ch
  12:   2a 2a                   sub    (%rdx),%ch
  14:   2a 0a                   sub    (%rdx),%cl
  16:   20 20                   and    %ah,(%rax)
  18:   20 2a                   and    %ch,(%rdx)
  1a:   0a 31                   or     (%rcx),%dh
  1c:   39 31                   cmp    %esi,(%rcx)
  1e:   32 32                   xor    (%rdx),%dh
  20:   30 30                   xor    %dh,(%rax)
  22:   30 38                   xor    %bh,(%rax)
  24:   0a 00                   or     (%rax),%al
```

真实学号：191220008

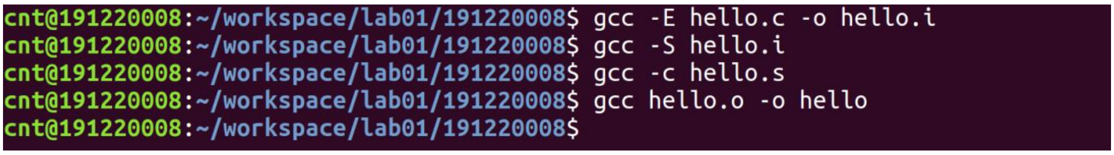
学号如下（除去 0a）



```
0a 31
39 31
32 32
30 30
30 38
```

即：31 39 31 32 32 30 30 30 38

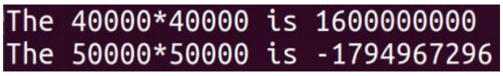
ii. 编写简单的 C 语言源程序 hello.c，通过预处理、编译、汇编、链接四个步骤将 C 语言源程序转换为可执行文件，即 hello.c -> hello.i -> hello.s -> hello.o -> hello



```
cnt@191220008:~/workspace/lab01/191220008$ gcc -E hello.c -o hello.i
cnt@191220008:~/workspace/lab01/191220008$ gcc -S hello.i
cnt@191220008:~/workspace/lab01/191220008$ gcc -c hello.s
cnt@191220008:~/workspace/lab01/191220008$ gcc hello.o -o hello
cnt@191220008:~/workspace/lab01/191220008$
```

4、数据的表示范围及不同类型的数据长度实验

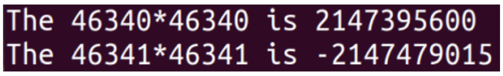
i. 将输出结果导出，说明发生这种现象的原因?



```
The 40000*40000 is 1600000000
The 50000*50000 is -1794967296
```

原因： int 类型整数对应的二进制位数是有上限的，当该整数超出2147483647 时会发生溢出，导致结果错误

ii. 寻找在该程序中保证结果正确的最大整数值?

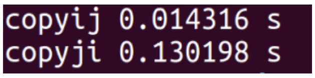


```
The 46340*46340 is 2147395600
The 46341*46341 is -2147479015
```

答案：46340

5、矩阵运算执行时间比较

i. 比较两个矩阵复制函数的执行时间。



```
copyij 0.014316 s
copyji 0.130198 s
```

ii. 说明为什么会出现这个差别。

原因：第一种复制函数按行进行读取（或者读取该行下一个），是连续访问内存；第二种复制函数按列进行读取，是跳跃访问内存。在进行数据缓存的时候是将二维数组的整行进行存储（其实不一定是整行，严格地说是存储一个区域的元素，只是在本题中 2048×2048 的数组中，因为元素较多，只能缓存部分元素，具体能存储多少元素，取决于数组大小、缓存大小、页面大小），如果按列读取则会去内存中读取。直接从缓存中读取数据的速度比从内存中读取更快。