Linux 简介

杨宗泽

September 8, 2021

历史

认识终端

目录结构

常用命令

正则表达式

作业调度系统 PBS

参考文献

Unix 和 Linux

年份	代表人物	操作系统	语言
1969	Ken Thompson	Unics(Unix)	汇编
1973	Dennis Ritchie	Unix	C
1984(1986)	Andrew Tanenbaum	Minix	C
1984	Richard Stallman	GNU (FSF 基金会)	
1991	Linus Torvalds	Linux	C

备注

- GNU (GUN is Not Unix)
- Ken Thompson 开发了 B 语言
- Dennis Ritchie 基于 B 语言创造了 C 语言
- Brian Kernighan 建议使用 Unix 这个名称代替 Unics
- Linux Torvalds 为了管理 Linux 内核的代码开发了 git 工具
- GNU 的软件有很多，如 GCC，GDB，GNU Emacs.

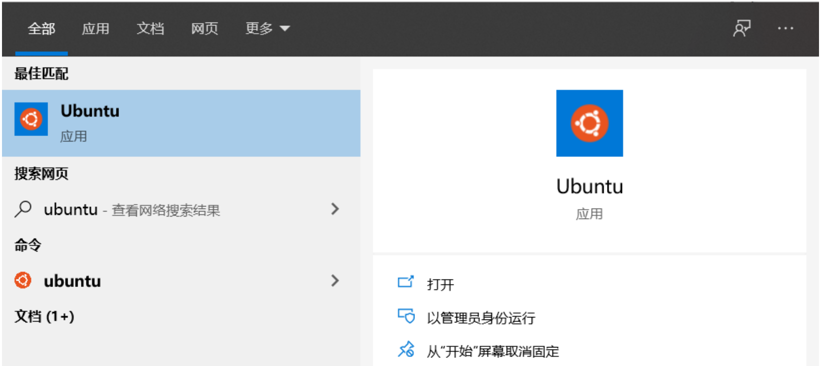
Linux 和 GNU 图标

Linux 常见发行版 Ubuntu, Debian, CentOS, Red Hat, ...

尝试 Linux

- Windows Linux Subsystem https://docs.microsoft.com/zh-cn/windows/wsl/install-win10
- 搞到某 Linux 服务器的账号
- 安装某 Linux 发行版 (如 Ubuntu)
- 购买云服务器 (阿里云, 华为云等)

终端 (Terminal)



全部 应用 文档 网页 更多

最佳匹配

Ubuntu
应用

搜索网页

ubuntu - 查看网络搜索结果

命令

ubuntu

文档 (1+)

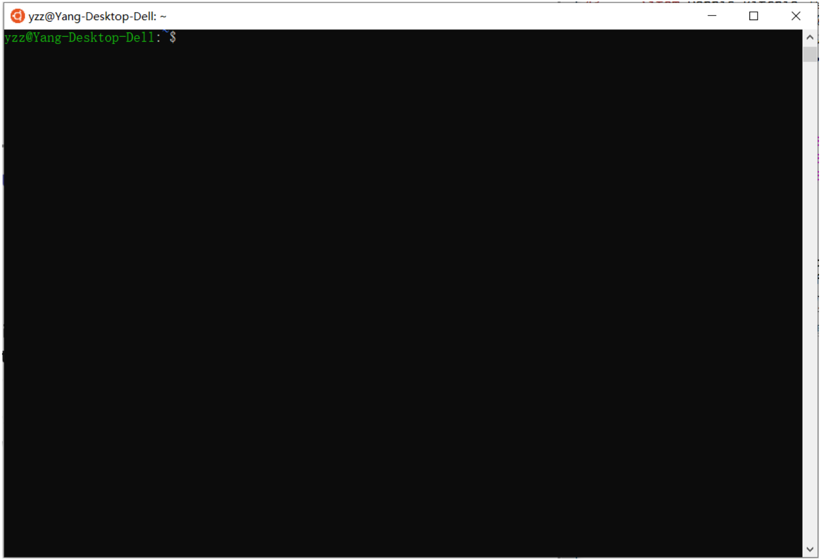
Ubuntu
应用

打开

以管理员身份运行

从"开始"屏幕取消固定

终端 (Terminal)



```
yzz@Yang-Desktop-Dell:~$
```

终端 (Terminal)

命令提示符 `yzz@Yang-Desktop-Dell:~$` 的说明

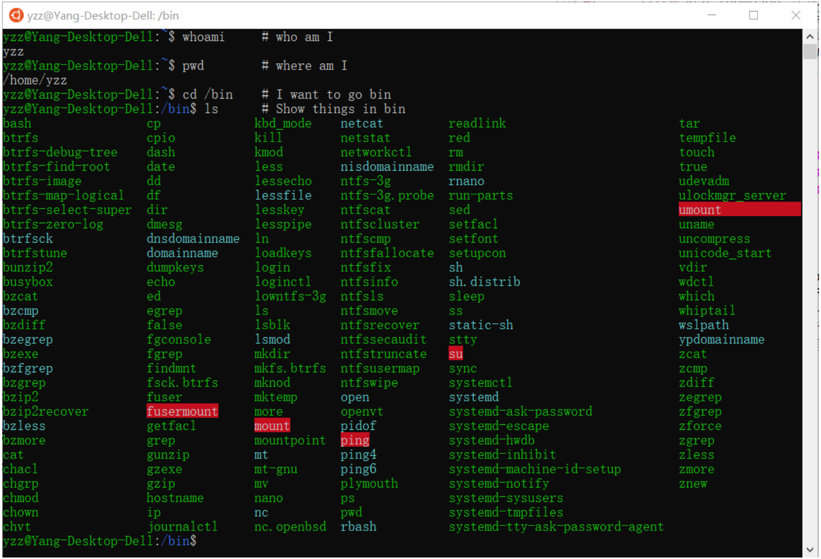
- `yzz`：用户名
- `Yang-Desktop-Dell`：主机名
- `~`：路径名，表示当前用户的 `home` 目录
- `$`：提示符

提示符一般的格式为

```
username@hostname:workdir\$
```

终端里运行的是被称作 Shell 的程序，负责用户和系统的交互.

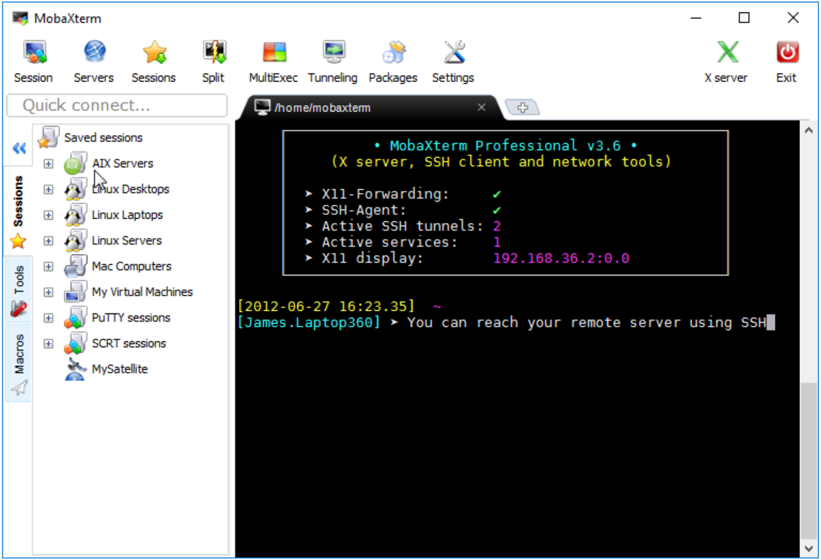
终端 (Terminal)



```
yzz@Yang-Desktop-Dell:~$ whoami     # who am I
yzz
yzz@Yang-Desktop-Dell:~$ pwd        # where am I
/home/yzz
yzz@Yang-Desktop-Dell:~$ cd /bin    # I want to go bin
yzz@Yang-Desktop-Dell:/bin$ ls      # Show things in bin
bash                cp             kbd_mode    netcat         readlink                        tar
btrfs               cpio           kill        netstat        red                             tempfile
btrfs-debug-tree    dash           kmod        networkctl     rm                              touch
btrfs-find-root     date           less        nisdomainname  rmdir                           true
btrfs-image         dd             lessecho    ntfs-3g        rnano                           udevadm
btrfs-map-logical   df             lessfile    ntfs-3g.probe  run-parts                       ulockmgr_server
btrfs-select-super  dir            lesskey     ntfscat        sed                             umount
btrfs-zero-log      dmesg          lesspipe    ntfscluster    setfacl                         uname
btrfsck             dnsdomainname  ln          ntfscmp        setfont                         uncompress
btrfstune           domainname     loadkeys    ntfsfallocate  setupcon                        unicode_start
bunzip2             dumpkeys       login       ntfsfix        sh                              vdir
busybox             echo           loginctl    ntfsinfo       sh.distrib                      wdctl
bzcat               ed             lowntfs-3g  ntfsls         sleep                           which
bzcmp               egrep          ls          ntfsmove       ss                              whiptail
bzdiff              false          lsblk       ntfsrecover    static-sh                       wslpath
bzegrep             fgconsole      lsmod       ntfssecaudit   stty                            ypdomainname
bzexe               fgrep          mkdir       ntfstruncate   su                              zcat
bzfgrep             findmnt        mkfs.btrfs  ntfsusermap    sync                            zcmp
bzgrep              fsck.btrfs     mknod       ntfswipe       systemctl                       zdiff
bzip2               fuser          mktemp      open           systemd                         zegrep
bzip2recover        fusermount     more        openvt         systemd-ask-password            zfgrep
bzless              getfacl        mount       pidof          systemd-escape                  zforce
bzmore              grep           mountpoint  ping           systemd-hwdb                    zgrep
cat                 gunzip         mt          ping4          systemd-inhibit                 zless
chacl               gzexe          mt-gnu      ping6          systemd-machine-id-setup        zmore
chgrp               gzip           mv          plymouth       systemd-notify                  znew
chmod               hostname       nano        ps             systemd-sysusers
chown               ip             nc          pwd            systemd-tmpfiles
chvt                journalctl     nc.openbsd  rbash          systemd-tty-ask-password-agent
yzz@Yang-Desktop-Dell:/bin$
```

默认终端的替代品

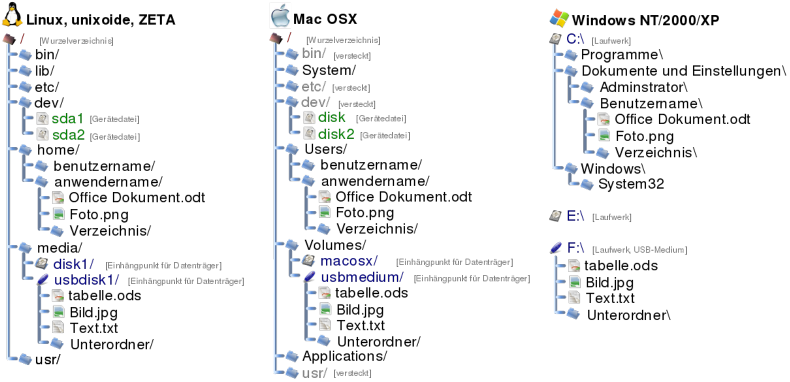
MobaXterm: https://mobaxterm.mobatek.net/



```
        • MobaXterm Professional v3.6 •
    (X server, SSH client and network tools)

  ➤ X11-Forwarding:      ✔
  ➤ SSH-Agent:           ✔
  ➤ Active SSH tunnels:  2
  ➤ Active services:     1
  ➤ X11 display:         192.168.36.2:0.0

[2012-06-27 16:23.35]  ~
[James.Laptop360] ➤ You can reach your remote server using SSH
```

Linux 目录结构

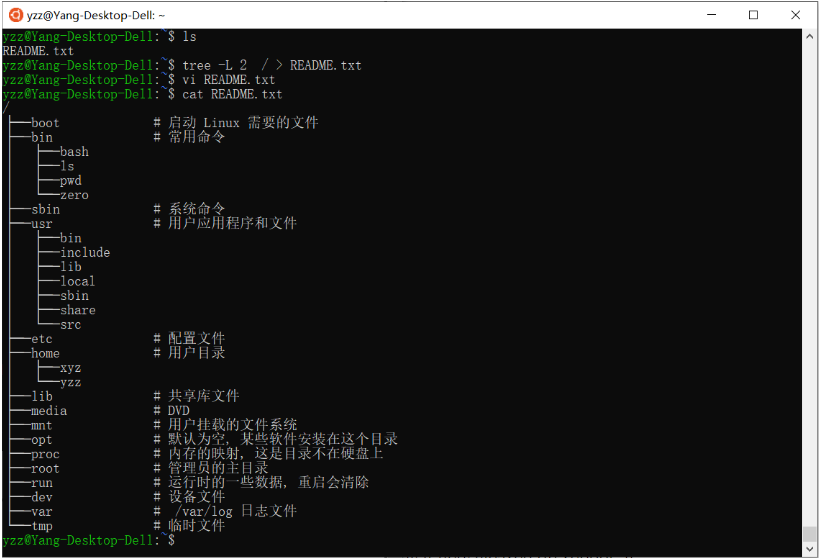
目录及结构对比 [1]



Linux/Unix：一切皆文件

[1] 图片来源：https://favpng.com/png_view/linux-file-system-directory-structure-computer-file-hierarchy-png/0mbj4crU

Linux 目录结构



```
yzz@Yang-Desktop-Dell:~$ ls
README.txt
yzz@Yang-Desktop-Dell:~$ tree -L 2  / > README.txt
yzz@Yang-Desktop-Dell:~$ vi README.txt
yzz@Yang-Desktop-Dell:~$ cat README.txt
/
├──boot             # 启动 Linux 需要的文件
├──bin              # 常用命令
│   ├──bash
│   ├──ls
│   ├──pwd
│   └──zero
├──sbin             # 系统命令
├──usr              # 用户应用程序和文件
│   ├──bin
│   ├──include
│   ├──lib
│   ├──local
│   ├──sbin
│   ├──share
│   └──src
├──etc              # 配置文件
├──home             # 用户目录
│   ├──xyz
│   └──yzz
├──lib              # 共享库文件
├──media            # DVD
├──mnt              # 用户挂载的文件系统
├──opt              # 默认为空，某些软件安装在这个目录
├──proc             # 内存的映射，这是目录不在硬盘上
├──root             # 管理员的主目录
├──run              # 运行时的一些数据，重启会清除
├──dev              # 设备文件
├──var              #  /var/log 日志文件
└──tmp              # 临时文件
yzz@Yang-Desktop-Dell:~$
```

常用命令 I

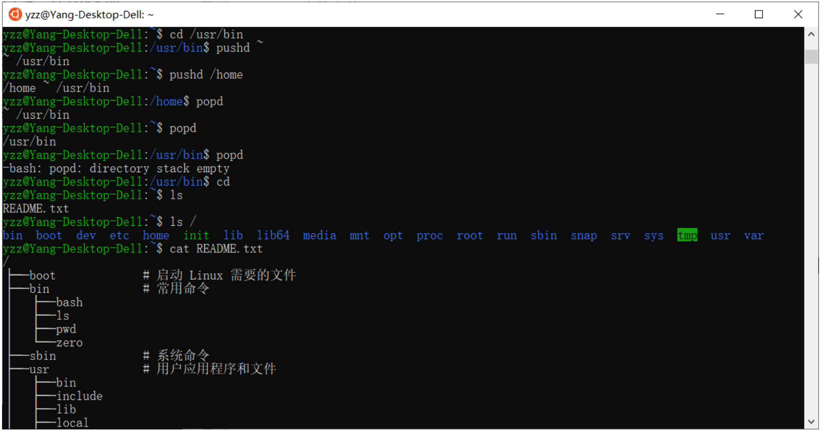
一些常用命令

分类	命令
切换目录	cd，pushd，popd
查看目录	ls，tree
搜索目录	find
编辑文件	vim，Emacs，nano
查看文件	cat，head，tail
查找文件	grep
处理文件	cut，sed，awk

一句话介绍

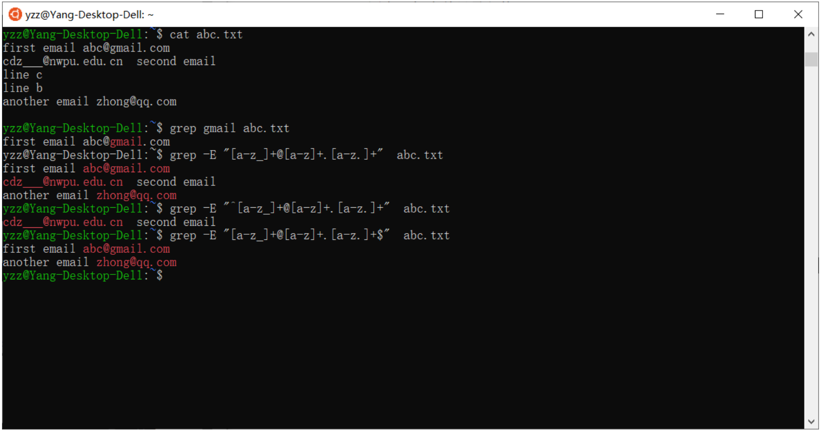
cd	切换目录
pushd	同上，把当前目录入栈
popd	切换目录，弹出栈顶
ls	查看目录下的文件
find	在目录中搜索文件
cat	合并文件并输出
grep	输出匹配某模式的行
cut	输出每行输入的某一部分

常用命令 II



```
yzz@Yang-Desktop-Dell:~$ cd /usr/bin
yzz@Yang-Desktop-Dell:/usr/bin$ pushd ~
~ /usr/bin
yzz@Yang-Desktop-Dell:~$ pushd /home
/home ~ /usr/bin
yzz@Yang-Desktop-Dell:/home$ popd
~ /usr/bin
yzz@Yang-Desktop-Dell:~$ popd
/usr/bin
yzz@Yang-Desktop-Dell:/usr/bin$ popd
-bash: popd: directory stack empty
yzz@Yang-Desktop-Dell:/usr/bin$ cd
yzz@Yang-Desktop-Dell:~$ ls
README.txt
yzz@Yang-Desktop-Dell:~$ ls /
bin  boot  dev  etc  home  init  lib  lib64  media  mnt  opt  proc  root  run  sbin  snap  srv  sys  tmp  usr  var
yzz@Yang-Desktop-Dell:~$ cat README.txt
/
├──boot              # 启动 Linux 需要的文件
├──bin               # 常用命令
│  ├──bash
│  ├──ls
│  ├──pwd
│  └──zero
├──sbin              # 系统命令
├──usr               # 用户应用程序和文件
│  ├──bin
│  ├──include
│  ├──lib
│  ├──local
```

常用命令 III

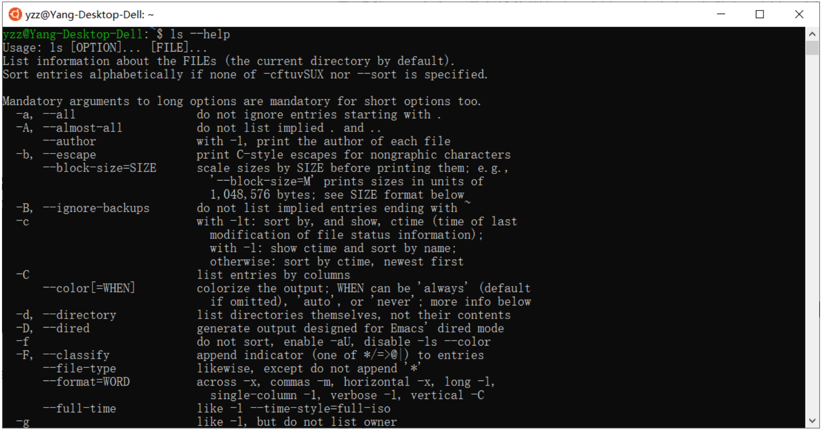


```
yzz@Yang-Desktop-Dell:~$ cat abc.txt
first email abc@gmail.com
cdz___@nwpu.edu.cn  second email
line c
line b
another email zhong@qq.com

yzz@Yang-Desktop-Dell:~$ grep gmail abc.txt
first email abc@gmail.com
yzz@Yang-Desktop-Dell:~$ grep -E "[a-z_]+@[a-z]+.[a-z.]+"  abc.txt
first email abc@gmail.com
cdz___@nwpu.edu.cn  second email
another email zhong@qq.com
yzz@Yang-Desktop-Dell:~$ grep -E "^[a-z_]+@[a-z]+.[a-z.]+"  abc.txt
cdz___@nwpu.edu.cn  second email
yzz@Yang-Desktop-Dell:~$ grep -E "[a-z_]+@[a-z]+.[a-z.]+$"  abc.txt
first email abc@gmail.com
another email zhong@qq.com
yzz@Yang-Desktop-Dell:~$
```

查看帮助 I

如何了解这些命令:

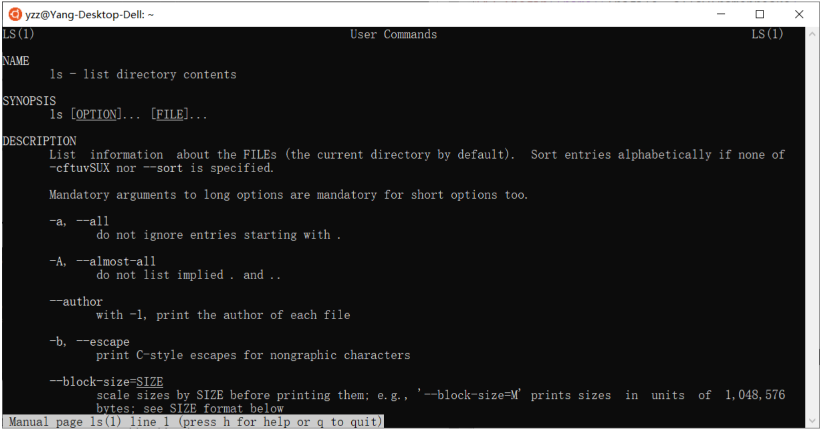
- 查看命令帮, `ls --help` 或者 `ls -h`



```
yzz@Yang-Desktop-Dell:~$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                  do not ignore entries starting with .
  -A, --almost-all           do not list implied . and ..
      --author               with -l, print the author of each file
  -b, --escape               print C-style escapes for nongraphic characters
      --block-size=SIZE      scale sizes by SIZE before printing them; e.g.,
                               '--block-size=M' prints sizes in units of
                               1,048,576 bytes; see SIZE format below
  -B, --ignore-backups       do not list implied entries ending with ~
  -c                         with -lt: sort by, and show, ctime (time of last
                               modification of file status information);
                               with -l: show ctime and sort by name;
                               otherwise: sort by ctime, newest first
  -C                         list entries by columns
      --color[=WHEN]         colorize the output; WHEN can be 'always' (default
                               if omitted), 'auto', or 'never'; more info below
  -d, --directory            list directories themselves, not their contents
  -D, --dired                generate output designed for Emacs' dired mode
  -f                         do not sort, enable -aU, disable -ls --color
  -F, --classify             append indicator (one of */=>@|) to entries
      --file-type            likewise, except do not append '*'
      --format=WORD          across -x, commas -m, horizontal -x, long -l,
                               single-column -1, verbose -l, vertical -C
      --full-time            like -l --time-style=full-iso
  -g                         like -l, but do not list owner
```

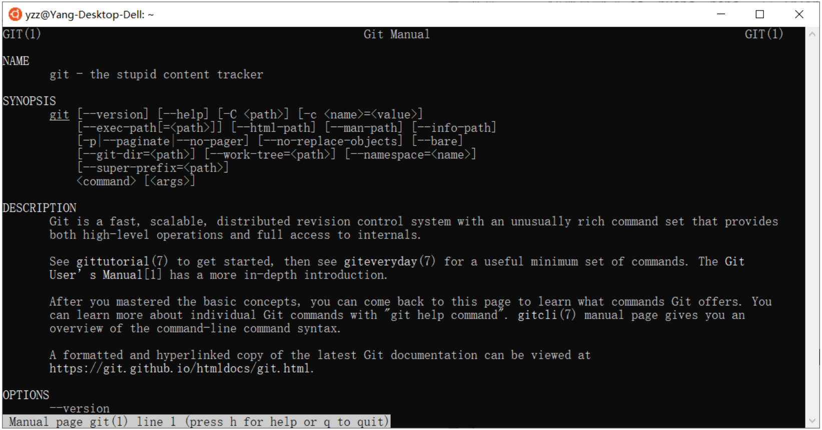
查看帮助 II

- 查看手册，如 `man ls`



```
LS(1)                                    User Commands                                   LS(1)

NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about the FILEs (the current directory by default).  Sort entries alphabetically if none of
       -cftuvSUX nor --sort is specified.

       Mandatory arguments to long options are mandatory for short options too.

       -a, --all
              do not ignore entries starting with .

       -A, --almost-all
              do not list implied . and ..

       --author
              with -l, print the author of each file

       -b, --escape
              print C-style escapes for nongraphic characters

       --block-size=SIZE
              scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes  in  units  of  1,048,576
              bytes; see SIZE format below
 Manual page ls(1) line 1 (press h for help or q to quit)
```

查看帮助 III

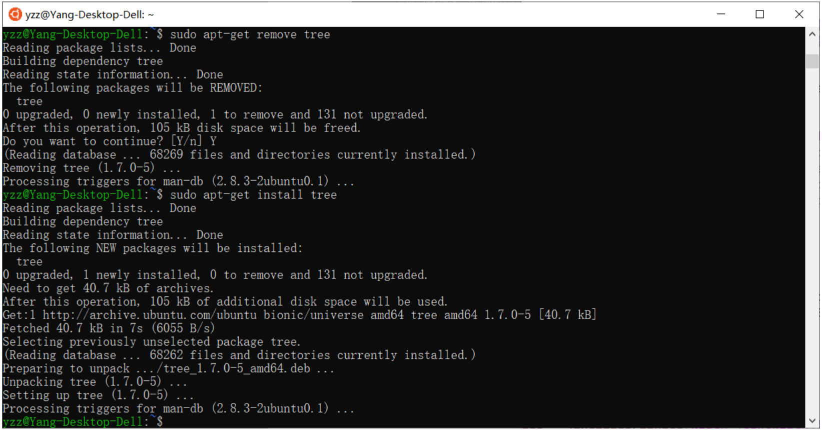


yzz@Yang-Desktop-Dell: ~

```
GIT(1)                                                  Git Manual                                                  GIT(1)

NAME
       git - the stupid content tracker

SYNOPSIS
       git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p|--paginate|--no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           [--super-prefix=<path>]
           <command> [<args>]

DESCRIPTION
       Git is a fast, scalable, distributed revision control system with an unusually rich command set that provides
       both high-level operations and full access to internals.

       See gittutorial(7) to get started, then see giteveryday(7) for a useful minimum set of commands. The Git
       User’ s Manual[1] has a more in-depth introduction.

       After you mastered the basic concepts, you can come back to this page to learn what commands Git offers. You
       can learn more about individual Git commands with "git help command". gitcli(7) manual page gives you an
       overview of the command-line command syntax.

       A formatted and hyperlinked copy of the latest Git documentation can be viewed at
       https://git.github.io/htmldocs/git.html.

OPTIONS
       --version
 Manual page git(1) line 1 (press h for help or q to quit)
```

安装软件

- 使用包管理器，Ubuntu 的 apt-get，CentOS 的 yum

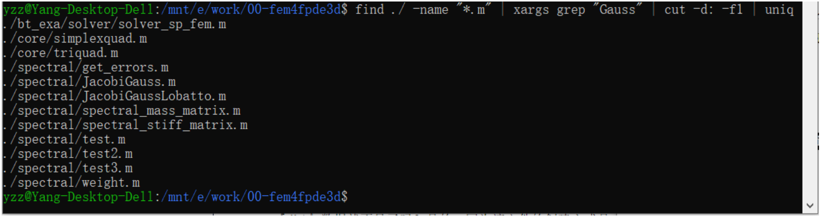


```
yzz@Yang-Desktop-Dell:~$ sudo apt-get remove tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  tree
0 upgraded, 0 newly installed, 1 to remove and 131 not upgraded.
After this operation, 105 kB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 68269 files and directories currently installed.)
Removing tree (1.7.0-5) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
yzz@Yang-Desktop-Dell:~$ sudo apt-get install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  tree
0 upgraded, 1 newly installed, 0 to remove and 131 not upgraded.
Need to get 40.7 kB of archives.
After this operation, 105 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic/universe amd64 tree amd64 1.7.0-5 [40.7 kB]
Fetched 40.7 kB in 7s (6055 B/s)
Selecting previously unselected package tree.
(Reading database ... 68262 files and directories currently installed.)
Preparing to unpack .../tree_1.7.0-5_amd64.deb ...
Unpacking tree (1.7.0-5) ...
Setting up tree (1.7.0-5) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
yzz@Yang-Desktop-Dell:~$
```

- 编译源码安装，阅读软件的 README.txt 和 Install 文件

示例

找出当前目录下所有内容包含"Gauss" 的"*.m" 文件.



```
yzz@Yang-Desktop-Dell:/mnt/e/work/00-fem4fpde3d$ find ./ -name "*.m" | xargs grep "Gauss" | cut -d: -f1 | uniq
./bt_exa/solver/solver_sp_fem.m
./core/simplexquad.m
./core/triquad.m
./spectral/get_errors.m
./spectral/JacobiGauss.m
./spectral/JacobiGaussLobatto.m
./spectral/spectral_mass_matrix.m
./spectral/spectral_stiff_matrix.m
./spectral/test.m
./spectral/test2.m
./spectral/test3.m
./spectral/weight.m
yzz@Yang-Desktop-Dell:/mnt/e/work/00-fem4fpde3d$
```

程序的 I/O 流[2]

- 标准输入流 (stdin, 0)
- 标准输出流 (stdout, 1)
- 标准错误流 (stderr, 2)

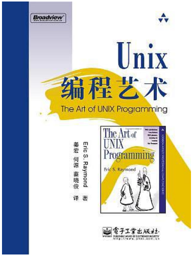
管道"|" 连通上一个程序的标准输入和下一个程序的输入.

[2] 和输入输出相关的另一个概念是重定向

管道概念中的 Unix 哲学

DO ONE THING AND DO IT WELL

- 一个程序只做一件事
- 程序设计的模块化
  一个函数或类只做一件事[3]
  - 命名要简洁有意义 (见名知义)
  - 函数要短小精悍 (只做一件事)
- 工作时一次做好一件事
  Do one thing at a time, and do it well

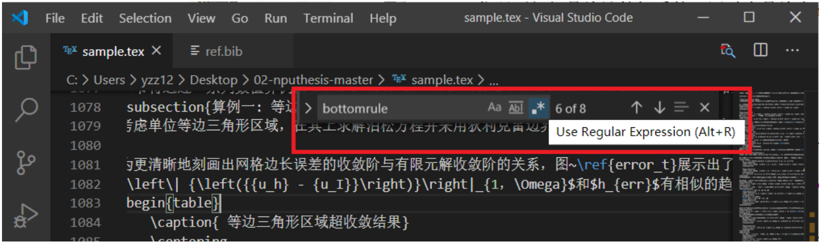


图灵程序设计丛书
Refactoring Improving the Design of Existing Code
重构 改善既有代码的设计
[美] Martin Fowler 著
熊节 译
软件开发的不朽经典
生动阐述重构原理和具体做法
普通程序员进阶到编程高手必须修炼的秘笈
人民邮电出版社

[1] Coding 建议: 良好的注释. 代码不仅仅是给计算机看的, 同时也是给人看的. (我们的目标是不写注释. 一周后: 这是啥? 谁写的? 写的啥?).

正则表达式 (Regular Expression)

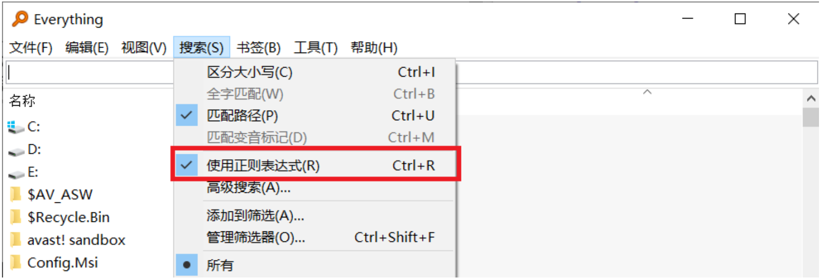
例子 `[a-z]+@[a-z]+.[a-z]+`



File
Edit
Selection
View
Go
Run
Terminal
Help
sample.tex - Visual Studio Code
sample.tex
ref.bib
C: > Users > yzz12 > Desktop > 02-nputhesis-master > sample.tex > ...
bottomrule
Aa
Abl
.*
6 of 8

Use Regular Expression (Alt+R)

```
subsection{算例一: 等边
虑单位等边三角形区域,

更清晰地刻画出网格边长误差的收敛阶与有限元解收敛阶的关系, 图~\ref{error_t}展示出了
\left\| {\left({{u_h} - {u_I}}\right)}\right|_{1, \Omega}$和$h_{err}$有相似的趋
begin{table}
    \caption{ 等边三角形区域超收敛结果}
```



正则表达式

正则表达式是一个字符串，定义了用于匹配某些字符串的规则.
这个字符串一般称作模式 (pattern).

模式	匹配的字符串	不匹配的字符串
a	a	b
a.b	aab, acb, a0b, a-b, ...	cab
a*b	b, ab, aab, ...	c, ac
a.*b	ab, acb, aab, ...	b, cb

点'.' 和'*' 具有特殊意义，叫做元字符

.	匹配任意一个字符.
[list]	匹配 list 中的人一个字符, 如 [abc] 匹配 a 或 b 或 c.
?	表示前面的字符重复至多 1 次.
*	表示前面的字符重复任意多次, 包括 0 次.
+	表示前面的字符至少重复一次, 如 a+ 匹配 aa 不匹配 a.
{m}	表示前面的字符匹配 m 次
{m,}	表示前面的字符匹配至少 m 次
{,n}	表示前面的字符匹配至多 m 次
{m,n}	表示前面的字符至少 m 次, 至多 n 次
^	匹配行首
$	匹配行尾

- 或: patternA | patternB
  gd|good 匹配 gd 或 good.
- 组: (pattern)
  g(la|oo)d 匹配 glad 或 good.

PBS

HPC 集群上的作业调度软件 PBS (Portable Batch System)

作业控制和监测:

qsub	提交作业
qdel	删除作业
qstat	显示作业状态

提交作业

```
qsub [options] <script.job>
```

作业文件示例

```
#!/bin/bash

### set job name
#PBS -N example-job
### set output files
#PBS -o example.stdout
#PBS -e example.stderr
### set queue name
#PBS -q example-queue
### set number of nodes
#PBS -l nodes=2:ppn=4

cd /home/yzz
cat README.txt
```

网络资源

- 鸟哥的私房菜 http://cn.linux.vbird.org/linux_basic/linux_basic.php
- 正则表达式 https://deerchao.cn/tutorials/regex/regex.htm
- Vim https://coolshell.cn/articles/5426.html

建议阅读章节 I

鸟哥的私房菜基础篇

- 第一章 Linux 是什么
  - 浏览.
- 第二章 Linux 如何学习
  - 浏览.
- 第五章首次登陆与在线求助
  - 学习 `man` 的用法
- 第七章 Linux 文件与目录管理
  - 了解 Linux 目录结构
  - 学习切换和查看目录: `cd`, `ls`
- 第九章文件与文件系统的压缩与打包
  - 学习 `tar` 命令
- 第十章 vim 程序编辑器
  - 学习 `vim` 基本操作命令
  - 或者看这里
  - 或者放弃本章, 尝试 `Emacs`.

建议阅读章节 II

- 第十一章认识与学习 BASH
  - 熟悉本章介绍的各种命令
- 第十二章正则表达式与文件格式化处理
  - 掌握正则表达式 (regular expression) 的基本语法
  - 也可以看这里
- 第十三章学习 Shell Scripts
  - 掌握脚本编写方法
- 第二十二章软件安装：原始码与 Tarball
  - 了解编译过程和 `make`
- 第二十三章软件安装：RPM, SRPM 与 YUM 功能
  - 了解软件安装方法

谢谢大家, 欢迎提问.