

Shell 其实就是一种脚本语言，也是一个可以用来连接内核和用户的软件，我们编写完源码后不用编译，直接运行源码即可。

bash, a general purpose scripting language.

command [arguments]

interactive and batch / 交互式和批处理

Gnuplot (在Linux下基于Qt的一套绘图工具)

https://blog.csdn.net/weixin\_42014622/article/details/82962634

LaTeX（LTEX，音译“拉泰赫”）是一种基于ΤΕΧ的排版系统

https://blog.csdn.net/pipisorry/article/details/54571521

关于shell 和 bash 之间的关系

<https://blog.csdn.net/weixin_42432281/article/details/88392219>

|  |  |  |
| --- | --- | --- |
| Platform | Linux | Windows |
| 字符界面模式 | shell | Command Prompt |
| 命令行 | bash | command |
| 批处理模式 | batch mode | 批处理 bat |

commands: aliases, functions, built-ins, keywords, and executables.

Text, letter

Description automatically generated

**Here document** 的使用总结

https://blog.csdn.net/liumiaocn/article/details/86715953

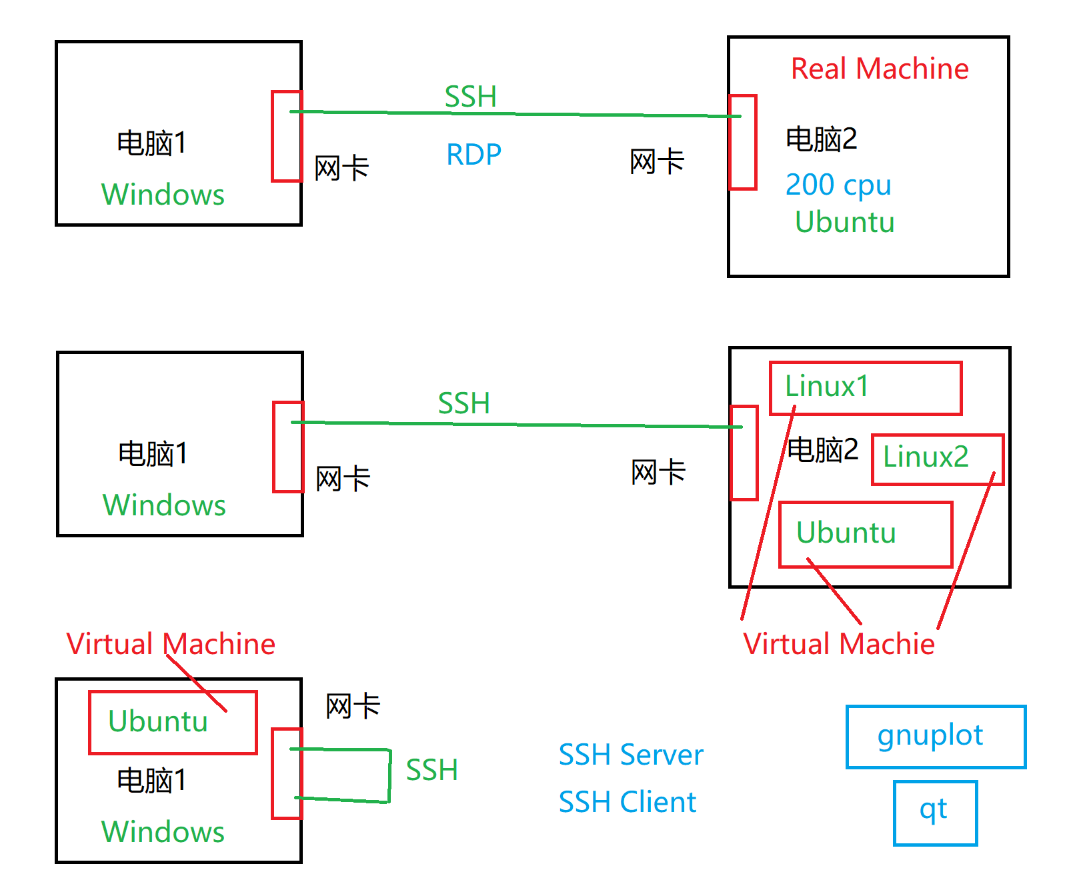
Sudo apt install / upgrade/update \*\*\*\*

Sudo apt-get install \*\*\*\*\*

确定型随机数是一个伪随机数，因为给定一相相同的种子seed，它就会生成相同的随机数，所以这种随机是假的，称为伪随机数。

如果想得到不同的随机序列，需要确保每次使用不同的种子，生成不同的序列。

Mersenne Twister算法译为马特赛特旋转演算法，是伪随机数发生器之一，其主要作用是生成伪随机数。



关于makefile和make:

<https://blog.csdn.net/ZBraveHeart/article/details/123187908>

Makefile文件中的命令有一定规范，一旦该文件编写好以后，在Linux命令行中执行一条make命令即可自动编译整个工程。

Text

Description automatically generated

它表示，为了达成目标，必须1，先有依赖，2，顺序执行下面的命令

Makefile并不会关心命令是如何执行的，仅仅只是会去执行所有定义的命令，和我们平时直接输入命令行是一样的效果。

1、目标即要生成的文件。如果目标文件的更新时间晚于依赖文件更新时间，则说明依赖文件没有改动，目标文件不需要重新编译。否则会进行重新编译并更新目标文件。

2、默认情况下Makefile的第一个目标为终极目标。

3、依赖：即目标文件由哪些文件生成。

4、命令：即通过执行命令由依赖文件生成目标文件。注意每条命令之前必须有一个tab保持缩进，这是语法要求（会有一些编辑工具默认tab为4个空格，会造成Makefile语法错误）。

5、all：Makefile文件默认只生成第一个目标文件即完成编译，但是我们可以通过all 指定所需要生成的目标文件。

**makefile中wildcard的理解**

<https://blog.csdn.net/dianqicyuyan/article/details/123509400>

wildcard 用来明确表示通配符。因为在 Makefile 里，变量实质上就是 C/C++ 中的宏，也就是说，如果一个表达式如 objs = \*.o ，则 objs 的值就是 \*.o ，而不是表示所有的 .o 文件。若果要使用通配符，那么就要使用 wildcard 来声明 \* 这个符号，使 \* 符号具有通配符的功能。

gnu make工具的[Makefile](https://so.csdn.net/so/search?q=Makefile&spm=1001.2101.3001.7020)中$(SRC:%.c=%.o)的含义：**将SRC变量中所有以.c结尾的文件名替换成对应的以.o结尾的文件名，然后赋回给SRC。**

[makefile](https://so.csdn.net/so/search?q=makefile&spm=1001.2101.3001.7020)**中.PHONY的作用是什么？**

<https://blog.csdn.net/weixin_41969690/article/details/106727385>

.PHONY all clean起到“欺骗”的作用，欺骗make别管别的，让你干嘛你就干嘛的意思。

缺省情况下，makefile中的target是一个文件，上面的定义就是告诉make命令，后面的target不是文件，而只是一个label而已

Makefile有三个非常有用的变量。分别是$@，$^，$<代表的意义分别是：

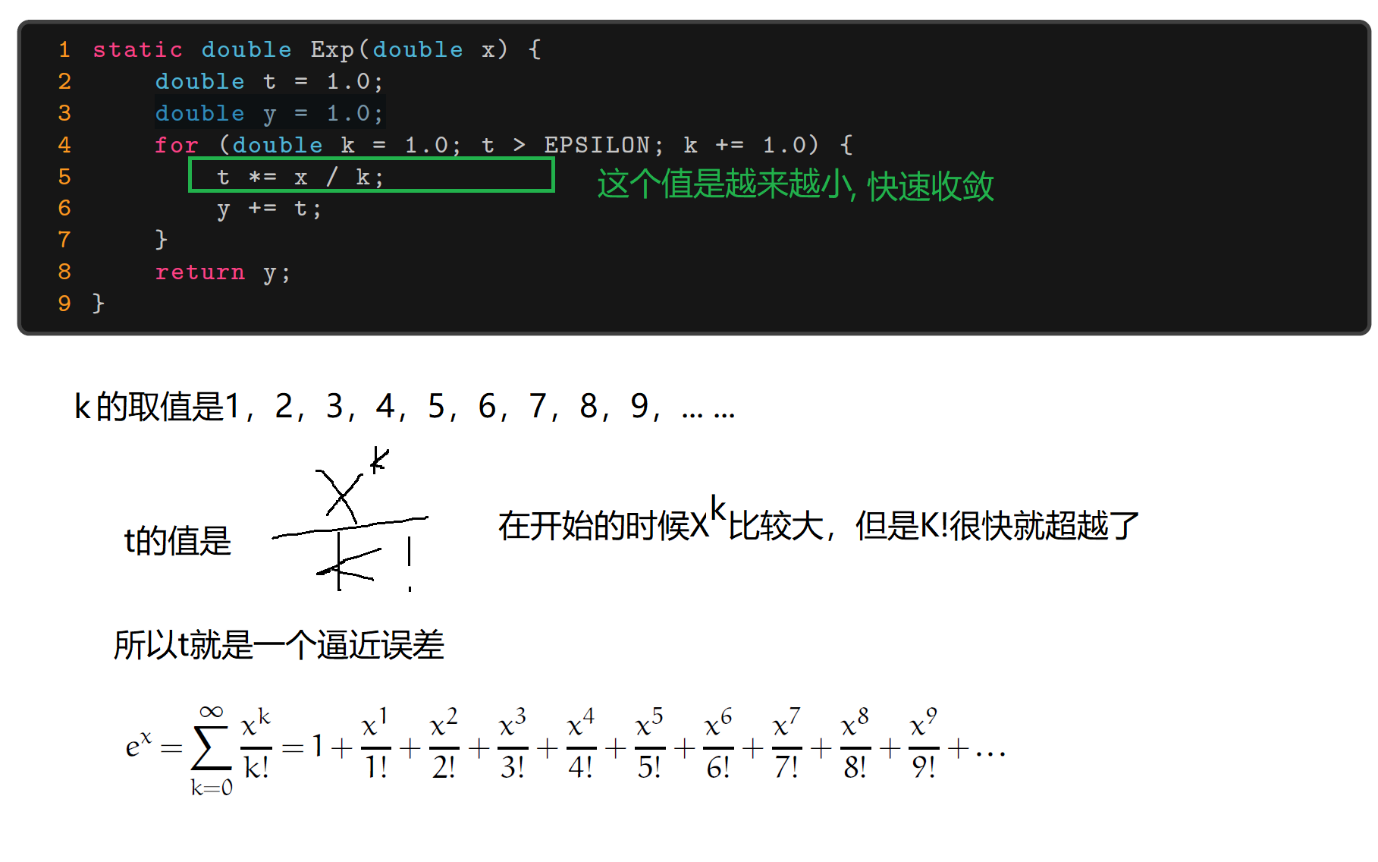
$@--目标文件，$^--所有的依赖文件，$<--第一个依赖文件。

mathlib.o : mathlib.c mathlib-test.c …

$< = mathlib.c

$@ = mathlib.o

$^ = mathlib.c mathlib-test.c …



Unix 文件系统，文件描述符（一个小的非负整数）

特别的，文件描述符0，1，2分别赋给stdin, stdout, stderr

Unix使用< 和 > 重定向和管道 | 改变输入/输出的IO

open 返回一个文件描述符

fopen 返回一个文件指针