

计算机基础

2017年3月14日 22:37

推荐书本：

css 权威指南（第三版）/css 程序设计

JS 权威指南

the book of css3

Eloquent JavaScript（适合入门）

我是一只IT小小鸟（多看网站）

快捷：

关闭窗口——左上角双击

在所在窗口目录打开命令行——shift+右键到在此处打开命令窗口

打开运行窗口——windows+r（在里面直接输入你要打开的运行程序名称即可快速打开）

原理：path, 路径，会去环境变量（计算机属性高级系统设置环境变量）依次匹配你输入的名称，加上扩展名（exe/com/bat/cmd/lnk）

path：是最常用的环境变量之一，它的值是一个目录列表当我们在运行窗口中输入命令的时候，系统会依次在这个目录列表内查找对应的文件名和扩展的文件，然后执行之。其扩展名的匹配范围是 exe/com/bat/cmd/lnk

一个使用场景是创建一个自己的文件夹然后将其添加到path列表里，然后在文件夹中创建自己常用的程序的快捷方式并命名为自己想要的，以后就可以直接在运行中输入快捷方式的文件名（不带扩展名）启动相应的程序了。

Everything安装

Strokit

费兹定律：一个目标的可达程序取决于他的距离和其大小，与距离呈反相关，与大小呈正相关。

具体应用：strokit, windows系统菜单设计。

操作系统：windows/os x(底层是unix)/Linux

浏览器：

IE->EDGE/Chrome/Firefox/Safari/Opera

主流浏览器内核：

（安卓系统内核 linux）

safari 内核：webkit

chrome 早期 webkit，现在blink

ie:trident

webkit Chrome35- Safari

blink Chrome35+ Opera32+
prostro Opera31-
gecko FireFox
Trident IE

壳浏览器 (IE+Chrome) : 百度 , 360 , qq浏览器等

移动端浏览器 :

壳浏览器、系统自带浏览器、微信浏览器、IOS系统App是不能自带浏览器内核的

兼容性 : 最低要支持IE9, 天猫已经放弃支持IE8

文本文件与二进制文件

所有的文件都是二进制文件, 如果有些文件被当成文本的内容来解析时, 所有的内容都是可识别以及可以被键盘输入的, 那我们一般称之为文本文件, html css js md都是文本文件。之所以区分是因为如果一个文件是文本文件, 那么我们可以用编辑器去方便的修改它。

二进制文件一般是使用16进制的形式查看。

命令行

2017年3月15日 14:07

GUI VS CLI

GUI:Graphics User Interface, 图形化**用户**界面/接口, 用户通过可视界面与应用进行交互

CLI:Command Line Interface, 命令行界面, 用户通过输入命令的形式与应用进行交互
接口/界面:与另一个事物进行交互的接口。

值得注意的是CLI不只限于Linux命令, 理论上任何应用都可以做成CLI形式, 如vi

具体应用如汽车的仪表盘、写信等

VPS:云主机 Virtual Private Server

API:Application Programming Interface 应用编程接口

在windows系统模拟linux环境的工具:

mingw

cgwin

安装git即自带上述其中一个工具。

一般命令:

ls:列出当前工作目录

pwd:打印出当前工作目录 (print working directory)

clear:清除当前窗口内容

带选项的命令:

ls -l:查看工作目录的详细信息

选项带值的命令:

1、翻墙:

退出所有谷歌浏览器, 然后在运行窗口里直接输入:

chrome.exe --proxy-pac-url="https://xrlovefruits.com:5050/pac.js"

或复制一个chrome在属性中修改目标为以上命令即可。

2、babel a.js -o a.compiled.js : 把a.js执行结果输出到a.compiled.js, -o为输出的意思

输出到文件:

echo abc > foo.txt :把命令的输出结果放到foo.txt文件

追加到文件

echo def >> foo.txt :文件若不存在则创建

pipe前一个的输出作为左后一个命令的输入：

管道符

echo def | xx | oo : 例如echo aaa| cat 输出aaa

常用linux命令：

ctrl+c:停止还没执行完的命令

cat a.txt:查看a文件的内容 (concatenate)

cat > b.txt:创建b.txt文件

(cat a.txt b.txt > aaa.txt :拼接命令,把a.txt与b.txt内容拼接成aaa.txt)

echo:回声,给什么输出什么

cd:相对路径/绝对路径, ('..'回到父目录, '.'当前目录) change directory

sudo:sudo mkdir dir

vi a.txt:虚拟编辑文本,ESC切换到命令模式,i切换到编辑模式,' :'完成命令,w保存,q退出,:wq保存并退出,q!强制退出

mkdir aaa :创建aaa文件夹

touch b.txt:创建文件

rmdir aaa:删除aaa文件夹

rm aaa/b.txt:删除aaa下的文件

rm -r a:删除a以及a下的所有文件,-r为递归的意思

cp a.txt c.txt:复制a文件到c文件

mv c.txt d.txt:重命名,把c变成d

time cp a.txt d.txt:显示运行该命令的时间

date:显示时间和日历

cal:显示日历

ping ip

便捷：

把命令行存到扩展名为sh的文件中,如复制文件等,直接执行即可,不需要每次操作

金条与二进制

2017年3月15日 14:08

金条与二进制问题，以加深对二进制的理解：

问题：干七天活，每天1块，一个金块值七块，要怎么切才可以每天给并且切的次数最少！

七天(7)

1 2 4 可以组成1-7 第一天给1，第二天给2找回1，第三天给1，第四天给4找回2、1，第五天给1，第六天给2找回1，第七天给1

若是半个月呢(15)

1 2 4 8 可以组成1-15

若是一个月呢(31)

1 2 4 8 16 可以组成1-31

规律1：31=2⁵-1，需要证明即2⁽ⁿ⁺¹⁾-1

规律2：1+2+4+8+16等于31，即等比数列求和 $1 \cdot ((1-2)^{(n+1)}) / 1-2$ 为 $2^{(n+1)}-1$

所以规律1成立！！

提问：1，2，4，8中组成和为9的组合有几种？貌似只有1，8

提问：1，2，4，8，16中组成和为12的组合有几种？貌似只有8，4

提问：1，2，4，8，...2的20次中组成120的组合有几种？

经理解十进制9的二进制为1001，12为1100

因此组成方式都是唯一的

图灵

2017年3月15日 14:09

图灵的基本思想是用机器来模拟人们用纸笔进行数学运算的过程，他把这样的过程看作下列两种简单的动作：

在纸上写上或擦除某个符号；

把注意力从纸的一个位置移动到另一个位置；

而在每个阶段，人要决定下一步的动作，依赖于（a）此人当前所关注的纸上某个位置的符号和（b）此人当前思维的状态。

为了模拟人的这种运算过程，图灵构造出一台假想的机器，该机器由以下几个部分组成：

1. 一条无限长的纸带 TAPE。纸带被划分为一个接一个小格子，每个格子上包含一个来自有限字母表的符号，字母表中有一个特殊的符号表示空白。纸带上的格子从左到右依此被编号为 $0, 1, 2, \dots$ ，纸带的右端可以无限伸展。
2. 一个读写头 HEAD。该读写头可以在纸带上左右移动，它能读出当前所指的格子上的符号，并能改变当前格子上的符号。
3. 一套控制规则 TABLE。它根据当前机器所处的状态以及当前读写头所指的格子上的符号来确定读写头下一步的动作，并改变状态寄存器的值，令机器进入一个新的状态。
4. 一个**状态寄存器**。它用来保存图灵机当前所处的状态。图灵机的所有可能状态的数目是有限的，并且有一个特殊的状态，称为停机状态。参见[停机问题](#)。

TAPE相当于电脑的内存，HEAD相当于电脑的cpu，Table相当于程序，符号表相当于ascii码（0-255）

计算机内部可以理解为：以8个格子为一组处理，8G内存就相当于 $8 \times 1024 \times 1024 \times 1024$ 个格子，每个格子里都是不同的二进制状态，一个格子为1bit，8个为1byte字节，所以一个字节可以表示256种状态，流量100M/s其实应该为 $100/8=12.5\text{M/s}$ ，即下载速度最多也就12.5M每秒。

计算机为什么要用二进制：如果用电压电流来表示肯定会造成一定的误差，用高电压（1）、低电压（0），产生的结果也是二进制，因此不会影响结果。

根本原因是因为模拟信号误差太大，必须想办法降低误差，甚至消除误差。能够控制误差范围的信号来表示0和1。

具体应用如猫、调制解调器即是用来转换网络中的数字信号与模拟信号的。

又如手机在播放音乐的时候，就是把MP3文件里的数字信息转换为扬声器里的模拟信号，即声音的震动。同样的如数字电视之类的。

收音机则是把接收到的模拟信号转换为扬声器里的声音，没有进行数模转换，因此会出现失真的状态。

模拟信号：如果我们把测试结果直接进行运算，就是在把信号当做模拟信号

数字信号：如果把测量结果看成是0/1，再把01参与运算，就是数字信号

markdown

2017年3月15日 14:06

在线编辑网站：<https://stackedit.io/editor> 或 <http://mahua.jser.me/>

编辑完成之后保存到本地，后缀为md,要想在本地浏览器打开，需要安装如下插件，安装完之后设置该插件允许访问网址打开即可，若出现乱码情况，需要设置其编码格式，即另存为utf-8编码格式即可。

<https://chrome.google.com/webstore/detail/github-flavored-markdown/fifkmhcgoobihkepohlhpkabplahglbo>

有道云笔记、github都支持markdown



markdown

编码

2017年3月15日 14:10

ASCII

enter(回车cr/换行lf) 10(mac)、13(linux)、1013(windows)

例子：你把linux很多换行的命令拷出来粘贴到windows下的记事本里，在记事本里只显示了一行，原因就是在linux系统里回车是13，而windows是1013，所以无法识别。因此可以选择一个非记事本的编辑器来保存，比如sublime就可以。

a 97
A 65
0 48
nbsp 160
space 32
cr (Carriage Return ,即enter) 13
lf (Line Feed , 即换行) 10

128个

几乎所有的编码方式都跟ascii兼容的，即在任何编码方式下，a的编号都是97

unicode

固定长，浪费空间

每8个字节表示一个unicode，一个字节8bit，所以 $8 \times 8\text{bit}$ ，即 2^6 次

一个字节(2^8)表示成16进制即为一个两位数，因为一个十六进制即 2^4 ，所以两个正好 2^8

范围是0x0000-0x10ffff， 16^6 次

只规定了不同符号的编号，具体在何种环境下如何表示取决于具体场景和编程语言。

(在ascii内的字符用utf8保存的时候，是一个字节一个字符)

utf-8 (只要保存成该格式，网站就能兼容各国语言)

变长，即不同的字符可能使用不同的存储空间来存放。而大多数的文件内容中英文(ascii)比较多，于是占用空间就会少一些，在网络中传输的流量就少。

自带纠错功能，丢失字节的话不会导致后面所有的内容乱码。

注：txt文件如果保存为utf-8格式，默认会增加文件头3个长度，所以就算内容只有一个a(一个英文字符占一个字节)，他的长度也是4！即ef bb bf 61 (16进制)

sublime就不会这样。

不要用utf-8 with bom,该编码方式等同记事本的utf-8 编码方式

BOM头：是用来指定文件是使用那种编码方式的，文件头长度不一定是3个长度。

编码方式：

变长编码，不同的unicode字符可能使用不同字节数量来保存

如果一个字符只需要一个字节保存，

0xxxxxxx

两个字节

110xxxxx 10xxxxxx

三个字节

1110xxxx 10xxxxxx 10xxxxxx

四

11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

码点的位数	码点起值	码点终值	字节序列	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
7	U+0000	U+007F	1	0xxxxxxx					
11	U+0080	U+07FF	2	110xxxxx	10xxxxxx				
16	U+0800	U+FFFF	3	1110xxxx	10xxxxxx	10xxxxxx			
21	U+10000	U+1FFFF	4	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx		
26	U+200000	U+3FFFFFF	5	111110xx	10xxxxxx	10xxxxxx	10xxxxxx	10xxxxxx	
31	U+4000000	U+7FFFFFFF	6	1111110x	10xxxxxx	10xxxxxx	10xxxxxx	10xxxxxx	10xxxxxx

如：“我” asc码为 25105，二进制即为“110001000010001”，因为15位，除以6即需要三个字节

所以为 1110【0110】 10【001000】 10【010001】

表示成十六进制即为 e6 88 91

```
> new Buffer("我")
<Buffer e6 88 91>
```

decode:

将百分号之间的16进制转换为二进制，查看前四位1的个数即表示该字符占的字节数，若没有1则直接转换为字符，否则按个数往后取若干个，后面的取其六位与之前的拼接最后得到的即为code

"abc.def%2F%2FE4%BD%A0%E6%88%91%2FE4%BB%96"<=>'abc.def//你/我/他'

base64编码 Data URI 多用于img标签

用于将不可打印的字符转换成可打印字符，或者简单的说将二进制数据编码成ASCII字符

基本上只有a-zA-Z0-9+ /这些字符，即64种，最后一般还会出现个等号

每四个字母在计算机中表示三个字节，每个位置有64种可能，即 2^6 ，所以四个是 2^{24} 次，三个字节每个 2^8 次，所以也是 2^{24} 次
即其四个6位二进制（通过查base64表）对应计算机中三个8位二进制

```

```

编码方式：将每个字节转为二进制，每6位分一组，从右往左分，每一组代表一个可打印字符

如：“我” asc码为 25105，二进制即为“110001000010001”，因为15位，除以6即需要三个字节

所以为 1110【0110】 10【001000】 10【010001】

转换为四个六位

111001 101000 100010 010001==》十进制即为 57, 40, 34, 17

查base64表得到 5oiR

```
> a=new Buffer("我")
<Buffer e6 88 91>
> a.toString("base64")
'5oiR'
```

浏览器中方法：atob() btoa() --只能操作英文数字等简单字符

node中方法：

buf=new Buffer("我")

buf.toString("base64")

特点：

对于小文件来说，直接嵌入会加快速度

没有缓冲，如果同个地址出现两次，会下载两次，而用url就只会下载一次，适合用在只出现一次的地方，如css的字体

如

```
@font-face {  
  font-family: yh;  
  src: url(data:xxxx);  
}
```

文件尺寸变为了原来的130%，体积变大

解码（查表，位运算）所需要的计算量较大，比直接解码二进制要大60%，不太适合在移动端用，比较费电费流量
老浏览器支持性不好