

批处理BAT脚本中for命令的使用



Batcher

2021年1月21日

在线视频分享约定

1. 专心听就行了，不用着急做笔记。视频和文档会放到网上供大家[免费下载](#)。
2. 请不要使用音箱等外放设备，以免有回声影响其他人，[最好使用耳机](#)。
3. 理论知识介绍完之后，随机从论坛上找几个帖子[现场演示](#)分析问题、解决问题的过程。

内容概要

- **for** 处理文件
- **for /d** 处理文件夹
- **for /r** 枚举目录树处理文件
- **for /l** 处理数字序列
- **for /f** 处理字符串、命令输出和文本内容

for 命令的基本语法

命令关键字：不区分大小写

for %%variable **in** (set) **do** command

控制变量：

- 区分大小写
- 推荐使用单个英文字母
- 不建议使用数字
- BAT脚本里面使用%%i
- CMD窗口里面使用%i
- 仅在for循环内部有效

文件集合：


- 一个或多个文件
- 可以使用通配符
- 需要放在圆括号里面
- 不判断文件是否真实存在

需要执行的命令：

- 多行命令需要放在圆括号里面
- BAT脚本里面使用
@echo off 之后不需要使用多余的@来关闭回显


for 命令处理文件

```
@echo off
for %%i in (a.txt "b c.txt") do (
    echo 正在处理: %%i
)
pause
```



```
C:\Test>test.bat
正在处理: a.txt
正在处理: "b c.txt"
请按任意键继续. . .
```

```
@echo off
for %%i in (*.txt ??.log) do (
    echo 正在处理: %%i
)
pause
```




```
C:\Test>test.bat
正在处理: a.txt
正在处理: b c.txt
正在处理: 1.log
正在处理: 20.log
请按任意键继续. . .
```

- 问号 ? 表示零个或一个任意字符。
- 星号 * 表示零个或一个或多个任意字符。
- 这里的问号 ? 和星号 * 是通配符，不是正则表达式。

for 命令的变量扩展


```
@echo off
for %%i in ("C:\Program Files\7-Zip\7z.exe") do (
    echo 文件: %%i
    echo 删除引号: %%~i
    echo 文件所在的驱动器: %%~di
    echo 文件所在的路径: %%~pi
    echo 文件名: %%~ni
    echo 文件扩展名: %%~xi
    echo 文件路径的短名: %%~si
    echo 文件的日期/时间: %%~ti
    echo 文件的大小: %%~zi
)
pause
```



文件: "C:\Program Files\7-Zip\7z.exe"
删除引号: C:\Program Files\7-Zip\7z.exe
文件所在的驱动器: C:
文件所在的路径: \Program Files\7-Zip\
文件名: 7z
文件扩展名: .exe
文件路径的短名: C:\PROGRA~1\7-Zip\7z.exe
文件的日期/时间: 2010/11/18 21:08
文件的大小: 284160
请按任意键继续. . .

```
@echo off
for %%i in ("C:\Program Files\7-Zip\7z.exe") do (
    echo 文件: %%i
    echo 文件所在的驱动器路径: %%~dpi
    echo 文件名和扩展名: %%~nxi
)
pause
```

➤ 在CMD窗口中执行 for /? 可以查看完整列表



文件: "C:\Program Files\7-Zip\7z.exe"
文件所在的驱动器路径: C:\Program Files\7-Zip\
文件名和扩展名: 7z.exe

for 命令的小技巧 (1)

❖ for命令不判断文件是否真实存在

```
@echo off
for %%i in (127.0.0.1) do (
    echo IP地址的前三段: %%~ni
)
pause
```

C:\Test>test.bat
IP地址的前三段: 127.0.0
请按任意键继续...

```
@echo off
for %%i in (3.1415926) do (
    echo 浮点数的整数部分: %%~ni
)
pause
```

C:\Test>test.bat
浮点数的整数部分: 3
请按任意键继续...

```
@echo off
for %%i in ("C:\Program Files\7-Zip") do (
    echo 目录的最后一级文件夹名称: %%~nxi
)
pause
```

C:\Test>test.bat
目录的最后一级文件夹名称: 7-Zip
请按任意键继续...

```
@echo off
for %%i in ("C:\Program Files\7-Zip\7z.exe\..") do (
    echo 文件所在的文件夹: %%~nxi
)
pause
```

C:\Test>test.bat
文件所在的文件夹: 7-Zip
请按任意键继续...

for 命令的小技巧 (2)

```
@echo off
for %%i in (test.bat) do echo %%~ni
```

- ❖ BAT脚本里面使用%%i
- ❖ BAT脚本里面使用 @echo off 之后不需要使用多余的@关闭回显

```
for %i in (test.bat) do @echo %~ni
```

- ❖ CMD窗口里面使用%i
- ❖ 命令前面可以使用@关闭回显

```
@echo off
for %%i in (*.txt) do (
    echo 正在处理: %%i
    ren "%%i" "new_%%i"
)
pause
```



- ❖ 小心文件集合重复处理的问题

```
C:\Test>dir /b *.txt
0_1.txt
0_2.txt
0_3.txt
```

```
C:\Test>test.bat
正在处理: 0_1.txt
正在处理: 0_2.txt
正在处理: 0_3.txt
正在处理: new_0_1.txt
请按任意键继续...
```

```
C:\Test>dir /b *.txt
new_0_2.txt
new_0_3.txt
new_new_0_1.txt
```

经常有人在论坛求助类似的问题

<http://bbs.bathome.net/thread-34670-1-1.html>

内容概要

- **for** 处理文件
- **for /d** 处理文件夹
- **for /r** 枚举目录树处理文件
- **for /l** 处理数字序列
- **for /f** 处理字符串、命令输出和文本内容

for /d 命令的基本语法

for /d %%variable in (set) do command

❖ 结合通配符处理文件夹

- 不处理文件
- 不枚举子文件夹
- 不处理隐藏文件夹

```
@echo off
for /d %%i in ("C:\Test\*") do (
    echo %%i
)
pause
```




```
C:\Test>test.bat
C:\Test\Folder1
C:\Test\Folder2
请按任意键继续. . .
```

```
C:\Test>attrib Folder3
H          C:\Test\Folder3
```

❖ 添加 /r 可以枚举子文件夹

```
@echo off
for /r "C:\Test\" /d %%i in (*) do (
    echo %%i
)
pause
```



```
C:\Test>test.bat
C:\Test\Folder1
C:\Test\Folder2
C:\Test\Folder1\Sub1_1
请按任意键继续. . .
```

内容概要


- **for** 处理文件
- **for /d** 处理文件夹
- **for /r** 枚举目录树处理文件
- **for /l** 处理数字序列
- **for /f** 处理字符串、命令输出和文本内容

for /r 命令的基本语法

for /r [[drive:]path] %%variable in (set) do command

❖ 使用星号枚举文件（遍历子文件夹）


```
@echo off
for /r "C:\Test\" %%i in (*.txt) do (
    echo %%i
)
pause
```



```
C:\Test>test.bat
C:\Test\0.txt
C:\Test\Folder1\1_1.txt
C:\Test\Folder1\Sub1_1\1_1_1.txt
C:\Test\Folder2\2_1.txt
请按任意键继续...
```

❖ 使用点号枚举文件夹（遍历子文件夹）

```
@echo off
for /r "C:\Test\" %%i in (.) do (
    echo %%i
)
pause
```



```
C:\Test>test.bat
C:\Test\
C:\Test\Folder1\
C:\Test\Folder1\Sub1_1\
C:\Test\Folder2\
C:\Test\Folder3\
请按任意键继续...
```

- 不处理隐藏文件和文件夹。
- 小心文件集合重复处理的问题。
- 如果 /r 后面不指定路径，默认处理当前文件夹。

for /r 命令的应用实例

```
@echo off
set "FileName=qq.exe"
set "AllDrive=C D E F G H I J K L M N O P Q R S T U V W X Y Z"
echo 正在全盘搜索%FileName%，请稍候...
for %%a in (%AllDrive%) do (
    if exist %%a:\ (
        pushd %%a:\
        for /r %%b in ("*%FileName%") do (
            if /i "%~nxb" equ "%FileName%" (
                echo,%%b
            )
        )
        popd
    )
)
pause
```

➤ echo命令和变量中间使用逗号，可以防止变量为空以及变量值是on或off的时候发生异常。

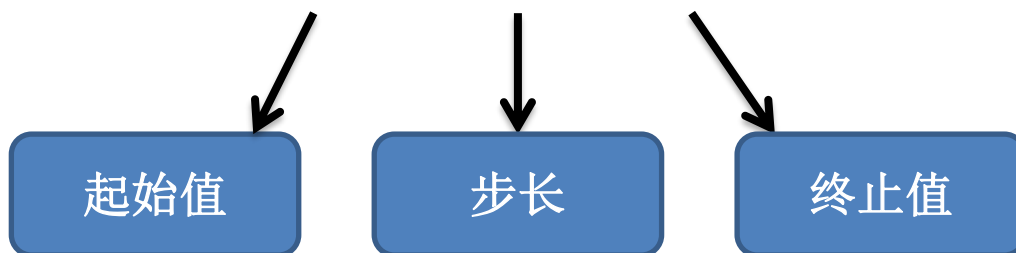
➤ 本文所有的脚本主要是为了展示语法，不一定是实现某个功能的最好的方法。

内容概要

- **for** 处理文件
- **for /d** 处理文件夹
- **for /r** 枚举目录树处理文件
- **for /l** 处理数字序列
- **for /f** 处理字符串、命令输出和文本内容

for /l 命令的基本语法

for /l %%variable in (start,step,end) do command



❖ 死循环	❖ 步长不一定是1	❖ 步长可以是负数	❖ 遍历完集合才退出
<pre>@echo off for /l %%i in () do (echo Hello) pause</pre>	<pre>@echo off echo 1到10之间的偶数: for /l %%i in (1,2,10) do (echo %%i) pause</pre>	<pre>@echo off echo 1到10倒序: for /l %%i in (10,-1,1) do (echo %%i) pause</pre>	<pre>@echo off for /l %%i in (1,1,999999) do (echo %%i if %%i equ 4 (goto :end)) :end pause</pre>

for /l 命令的应用实例

@echo off

REM 输出九九乘法表

```
for /l %%i in (1,1,9) do (  
    setlocal enabledelayedexpansion  
    for /l %%j in (1,1,%%i) do (  
        set /a n=%%j*%%i  
        set /p =%%j*%%i=!n! <nul  
    )  
endlocal  
echo,  
)  
pause
```

➤ 使用 set /p 可以实现不换行输出

<http://bbs.bathome.net/viewthread.php?tid=3774#pid23888>

➤ 在真正需要的时候才打开变量延迟扩展

➤ 及时关闭变量延迟扩展（尤其是在for循环内部）

<http://bbs.bathome.net/thread-1224-1-1.html>

➤ 使用echo命令输出空行的方法

<http://bbs.bathome.net/thread-13860-1-1.html>

内容概要

- **for** 处理文件
- **for /d** 处理文件夹
- **for /r** 枚举目录树处理文件
- **for /l** 处理数字序列
- **for /f** 处理字符串、命令输出和文本内容

for /f 命令的基本语法

for /f ["options"] %%variable in (file-set) do command



文件集合

for /f ["options"] %%variable in ("string") do command



字符串

for /f ["options"] %%variable in ('command') do command



命令的执行结果

for /f 命令的选项 eol

中文版系统帮助文档太(Zhen)任 (Keng)性(Die)

eol=c

- 指一个行注释字符的结尾(就一个)
- specifies an end of line comment character (just one)

@echo off

REM 默认忽略分号开头的行

```
for /f %i in (test.txt) do (  
    echo,%i  
)  
pause
```

C:\Test>type test.txt
;第1行默认注释
#第2行自定义注释
第3行正文

C:\Test>test.bat
#第2行自定义注释
第3行正文
请按任意键继续. . .

@echo off

REM 忽略指定字符开头的行

```
for /f "eol=#" %i in (test.txt) do (  
    echo,%i  
)  
pause
```

C:\Test>type test.txt
;第1行默认注释
#第2行自定义注释
第3行正文

C:\Test>test.bat
;第1行默认注释
第3行正文
请按任意键继续. . .


- 所谓的 **end of line comment** 是一个专有名词，它是一种注释方法。意思是，如果某行以指定字符开头，那么整行都是注释。比如 **Linux Shell** 脚本里面的井号，**VBS** 脚本里面的单引号，等等。与之相对应的，是类似 **C** 语言里面的 `/* xyz */` 这种可以放到一行代码中间的注释方法。

for /f 命令的选项 skip


skip=n

- 指在文件开始时忽略的行数。

```
@echo off
REM 跳过前两行
for /f "skip=2" %%i in (test.txt) do (
    echo,%%i
)
pause
```



```
C:\Test>type test.txt
第1行
第2行
第3行
第4行
第5行
```




```
C:\Test>test.bat
第3行
第4行
第5行
请按任意键继续. . .
```


```
@echo off
REM 把两个文件逐行合并成一行
set n=0
for /f %%i in (a.txt) do (
    call :MergeColumn %%n%% %%i
    set /a n+=1
)
pause
goto :eof
```

```
:MergeColumn
if %1==0 (
    set "myskip="
) else (
    set "myskip=skip=%1"
)
for /f "%myskip%" %%j in (b.txt) do (
    echo,%2 %%j
    goto :eof
)
```

```
C:\Test>type a.txt
文件a第1行
文件a第2行
文件a第3行
```



```
C:\Test>type b.txt
文件b第1行
文件b第2行
文件b第3行
```



```
C:\Test>test.bat
文件a第1行 文件b第1行
文件a第2行 文件b第2行
文件a第3行 文件b第3行
请按任意键继续. . .
```

for /f 命令的选项 tokens

tokens=x,y,m-n

- 指定需要获取的列

```
C:\Test>type a.txt
```

```
A B C D E F G H I J K L M N
```

```
@echo off
```

```
REM 默认获取第一列  
for /f %%i in (a.txt) do (
```

```
    echo,%%i
```

```
)
```

```
pause
```

```
@echo off
```

```
REM 逗号间隔，减号表示范围
```

```
for /f "tokens=1,3,5-7" %%i in (a.txt) do (
```

```
    echo,%%i %%j %%k %%l %%m
```

```
)
```

```
pause
```

```
@echo off
```

```
REM 星号表示后面的所有列
```

```
for /f "tokens=1,*" %%i in (a.txt) do (
```

```
    echo,%%i
```

```
    echo,%%j
```

```
)
```

```
pause
```

```
C:\Test>test.bat
```

```
A  
请按任意键继续. . .
```

```
C:\Test>test.bat
```

```
A C E F G  
请按任意键继续. . .
```

```
C:\Test>test.bat
```

```
A  
B C D E F G H I J K L M N  
请按任意键继续. . .
```

- 列很多的情况: <http://bbs.bathome.net/thread-5548-1-1.html>
- 获取最后一列: <http://bbs.bathome.net/thread-13000-1-1.html>

for /f 命令的选项 delims

delims=xxx

- 指定分隔符集。用来替换空格和跳格键（制表符）的默认分隔符集。
- **tokens** 和 **delims** 同时出现时，建议把 **delims** 放在 **tokens** 后面。

```
C:\Test>type a.txt
```

```
A B C:D:E F G+H-I ABC M N ABC X Y Z
```

```
@echo off
```

```
REM 默认分隔符是空格和制表符
```

```
for /f "tokens=3" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat
```

```
C:D:E  
请按任意键继续. . .
```

```
@echo off
```

```
REM 设置加号和减号为分隔符（如有空格请放在最后）
```

```
for /f "tokens=3 delims=+-" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat
```

```
I ABC M N ABC X Y Z  
请按任意键继续. . .
```

```
@echo off
```

```
REM 设置冒号为分隔符
```

```
for /f "tokens=3 delims=:" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat
```

```
E F G+H-I ABC M N ABC X Y Z  
请按任意键继续. . .
```

```
@echo off
```

```
REM 不要企图把字符串作为一个整体分隔符
```

```
for /f "tokens=3 delims=ABC" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```



```
C:\Test>test.bat
```

```
:D:E F G+H-I  
请按任意键继续. . .
```

for /f 命令tokens和 delims 小技巧

```
C:\Test>type a.txt  
A B C"D E F"G
```

```
    a b c"d e f"g  
1 2 3"4 5 6"7
```

```
@echo off  
REM 删除空行并且删除行首的空格或制表符  
for /f "tokens=*" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat  
A B C"D E F"G  
a b c"d e f"g  
1 2 3"4 5 6"7  
请按任意键继续. . .
```

```
@echo off  
REM 删除空行但是不删除行首的空格或制表符  
for /f "delims=" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat  
A B C"D E F"G  
    a b c"d e f"g  
1 2 3"4 5 6"7  
请按任意键继续. . .
```

```
@echo off  
REM 以双引号作为列分隔符  
for /f tokens^=2^ delims^=" %%i in (a.txt) do (  
    echo,%%i  
)  
pause
```

```
C:\Test>test.bat  
D E F  
d e f  
4 5 6  
请按任意键继续. . .
```

for /f 命令的选项 usebackq

usebackq

- 允许在 file-set 中使用双引号扩起文件名称。

```
C:\Test>type "a 1.txt"
```

```
A B C D E F G
```

```
@echo off
REM 默认把双引号里面的内容当做字符串处理
for /f "tokens=2" %%i in ("a 1.txt") do (
    echo,%%i
)
pause
```

```
C:\Test>test.bat
```

```
1.txt
```

```
请按任意键继续. . .
```

```
@echo off
REM 正确处理文件名包含空格的情况
for /f "usebackq tokens=2" %%i in ("a 1.txt") do (
    echo,%%i
)
pause
```

```
C:\Test>test.bat
```

```
B
```

```
请按任意键继续. . .
```

➤ 极少用到的 usebackq 知识点（不建议使用这种写法）：

1. 原来的 ("string") 要把双引号改为单引号 ('string')
2. 原来的 ('command') 要把单引号改为后引号 (`command`)

for /f 命令里面特殊字符要转义

❖ 管道符号、等号、逗号、重定向符号等特殊字符需要转义

```
@echo off
for /f "delims=" %%i in ('dir /b *.txt ^| findstr "[0-9]"') do (
    echo,%%i
)
pause
```

<http://bbs.bathome.net/thread-1290-2-1.html#pid6306>

```
@echo off
for /f "delims=" %%i in ('wmic LogicalDisk where DriveType^="3" get DeviceID') do (
    echo,%%i
)
pause
```

<http://bbs.bathome.net/thread-56922-1-1.html#pid232844>

```
@echo off
for /f "delims=" %%i in ('wmic NicConfig get IPAddress^, MACAddress /value') do (
    echo,%%i
)
pause
```

<http://bbs.bathome.net/thread-56922-1-1.html#pid232844>

```
@echo off
for /f "delims=" %%i in ('dir /b /s /a-d "*.jpg" 2^>nul') do (
    echo %%i
)
pause
```

<http://bbs.bathome.net/thread-57460-1-1.html#pid234531>

小心处理文件扩展名

❖ 扩展名可能让你处理到本来不想处理的文件

```
@echo off
for /f "delims=" %%i in ('dir /b "*.htm"') do (
    echo,%%i
)
pause
```



```
C:\Test>test.bat
1.htm
2.html
请按任意键继续. . .
```

```
@echo off
for /f "delims=" %%i in ('dir /b "*.htm" ^| findstr "htm$"') do (
    echo,%%i
)
```



```
C:\Test>test.bat
1.htm
请按任意键继续. . .
```

实际例子: <http://bbs.bathome.net/thread-32507-1-1.html>

findstr正则表达式用法: <http://bbs.bathome.net/thread-24-1-1.html>

不要在 for 命令内部定义标签

❖ 在 for 命令内部定义的标签往往不会按照你预期的流程去执行

```
@echo off
for %%i in (1.txt 2.txt) do (
    goto :show

    :show
    echo 正在处理 %%i
)
pause
```



```
C:\Test>test.bat
正在处理 %i
请按任意键继续. . .
```

```
@echo off
for %%i in (1.txt 2.txt) do (
    call :show %%i
)
pause
goto :eof

:show
echo 正在处理 %1
```



```
C:\Test>test.bat
正在处理 1.txt
正在处理 2.txt
请按任意键继续. . .
```

实际例子:

<http://bbs.bathome.net/thread-28181-1-1.html>

<http://bbs.bathome.net/thread-29009-1-1.html>

for 控制变量仅在循环内部有效

```
@echo off
for /f %%i in ('set ^| findstr "ComSpec"') do (
    echo %%i
)
echo %%i
pause
```



for循环结束之后，不能在for循环外面直接使用控制变量的值。

```
@echo off
for /f %%i in ('set ^| findstr "ComSpec"') do (
    echo %%i
    set str=%%i
)
echo %str%
pause
```



如果需要在for循环外面对变量做进一步处理，可以把控制变量赋值给一个普通变量，然后在for循环外面使用这个普通变量进行后续操作。

练习题

<http://bbs.bathome.net/thread-2336-1-1.html>

<http://bbs.bathome.net/thread-459-1-1.html>

<http://bbs.bathome.net/thread-850-1-1.html>

- 多做练习题熟悉语法是好事但是没有必要过于沉迷
- 多去论坛尝试帮助别人解决实际问题
- 今天别人求助的问题明天就可能出现在你自己的工作中

THANKS
&
GOOD LUCK