

导言

这里存放一些实际工作中碰到的实用的代码片段，可能包含 MATLAB、Python、C 和一些 \LaTeX 的小知识。

第一章 MATLAB

问题 1 如何遍历当前文件夹及其子文件夹中的全部文件

假设现在我们有这样一个文件夹 A，它含有一些文件和子文件夹 B、C、D..... 我们需要将这个父文件夹（A）及其子文件夹（B、C、D.....）中的所有文件名和其路径取出来。

如果你用的是 MATLAB 2016b 及其后面的版本，那真的太棒了！`dir()`函数已经支持遍历搜索了。尝试敲入，

```
dir_data = dir('**/*');  
dir_data([dir_data.isdir]) = []; % 去除所有文件夹
```

这将会返回一个包含文件信息的 struct，现在你可以任意操作这些 struct 了，随意拼接路径。解放大脑，哦也！

方便归方便，但是，一来肯定有大多数人使用的是 MATLAB 2016b 之前的版本，二来，解放大脑意味着我失去了一次独立思考的机会。

对于实现方法，多层次的遍历，我们第一时间想到的肯定是递归。其次是数据的存储，`dir()`函数返回的是一个 struct，我们要充分利用这个数据结构。所以现在思路是，写一个递归函数，这个函数返回包含所以文件的 struct。

这个函数应对先处理父文件夹，获取文件和文件夹，然后获取子文件夹，我们对获取的子文件夹再次调用该函数，最后函数返回存储有所有文件信息的 struct。现在，你可以对这个结构体做你想做的事情。

```
function file_list = get_all_file_name(path)  
    dir_data = dir(path);  
    file_list = dir_data(~[dir_data.isdir]); % 当前文件夹下所有文件  
  
    % 获取子文件夹  
    sub_dir = dir_data([dir_data.isdir]); % struct  
    dot_dir = ismember({sub_dir.name}, {'.', '..'}); % logical  
    sub_dir = sub_dir(~dot_dir); % struct, 不含特殊文件夹的所有子文件夹  
  
    % 递归  
    for i = 1:length(sub_dir)  
        next_dir = fullfile(sub_dir(i).folder, sub_dir(i).name); % str  
        file_list = [file_list; get_all_file_name(next_dir)]; % struct  
    end  
end
```

1.1.1 关于效率

`dir()`函数遍历整个F盘文件大约需要1.555823s。我们实现的递归函数遍历F盘文件大约需要3.703009s。慢是慢了点，但我们成功运用了递归解决问题，不是吗？