网络爬虫

1. **对网页的遍历**

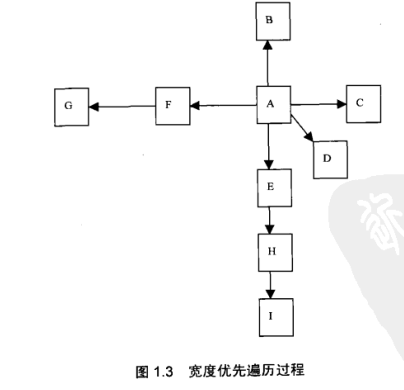
对于网页的遍历就好像我们对于图的遍历一样，每个网页下面有很多超链接，我们以一个网页作为起始节点进行遍历。

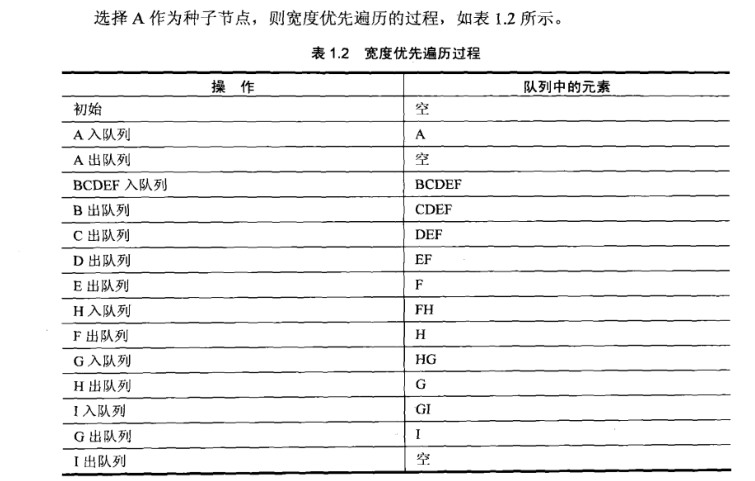
其中一共分为两种遍历

深度遍历：深度遍历网页容易陷入遍历黑洞，不推荐使用

宽度遍历：推荐使用

1. **宽度遍历**





1. **Java实现**

**本质：**爬虫的本质就是将获取到的html文件或者json格式的数据或者js文件的内容解析出我们需要的数据。

**实现：**

**public** **static** **int** getBrand(**int** pageCount,**int** moduleId){

//设置请求的url地址

String strUrl = "http://hqdigi2.eastmoney.com/EM\_Quote2010NumericApplication/index.aspx?type=s&sortType=C&sortRule=-1&pageSize=20&page="

+pageCount+"&jsName=quote\_123&style=28002"+moduleId+"&token=44c9d251add88e27b65ed86506f6e5da";

//建立连接

Connection connection = Jsoup.*connect*(strUrl);

//设置http响应头

connection.userAgent("Mozilla/5.0 (Windows NT 5.1; rv:33.0) Gecko/20100101 Firefox/33.0");

connection.ignoreContentType(**true**);

connection.ignoreHttpErrors(**true**);

Response response = **null**;

**try** {

response = connection.execute();

//获取到请求返回的字符串

String result = **new** String(response.bodyAsBytes(),"utf-8");

/\*----------解析获取到的字符串---------------\*/

//解析出数据的页数

String[] str = result.split("\"\\],pages:");

String pageCountStr = str[1];

pageCountStr = pageCountStr.split("}")[0];

pageCount = Integer.*parseInt*(pageCountStr);

String dataStr = str[0].split("rank:\\[\"")[1];

String[] dataArr = dataStr.split("\",\"");

BrandDbUtil brandDbUtil = **new** BrandDbUtil();

**for** (**int** i = 0; i < dataArr.length; i++) {

String[] data = dataArr[i].split(",");

Brand brand = **new** Brand();

brand.setName(data[2]);

brand.setModuleId(moduleId);

brandDbUtil.insertData(brand);

}

brandDbUtil.closeStatement();

} **catch** (IOException e) {

e.printStackTrace();

}

**return** pageCount;

}

1. **爬虫url参数**

问题描述：用Java程序去获取网络请求的内容，url和请求参数都没有问题，cookie也是对的，但是就是获取不到请求的内容。

解决方法：这个时候需要检查请求的URL中是否带有中文参数，一般情况下对于中文参数都是要进行URL编码的。

比如：[http://www.uniocredit.com/companyQuery.action?provice=BJ&key=易宝&name=北京](http://www.uniocredit.com/companyquery.action?provice=bj&key=%25e6%2598%2593%25e5%25ae%259d&name=%25e5%258c%2597%25e4%25ba%25ac)编码后成了

[http://www.uniocredit.com/companyQuery.action?provice=BJ&key=%E6%98%93%E5%AE%9D&name=%E5%8C%97%E4%BA%AC](http://www.uniocredit.com/companyquery.action?provice=bj&key=%25e6%2598%2593%25e5%25ae%259d&name=%25e5%258c%2597%25e4%25ba%25ac)

但是有的时候我们进行URL编码再去请求可能获取不到请求内容，因此这个时候就不必对其进行编码。

1. **网站防扒策略**

第一、根据ip来限制一段时间段的请求次数

第二、同一个请求短时间内只能发送一次

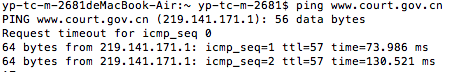
第三、

1. **请求超时**

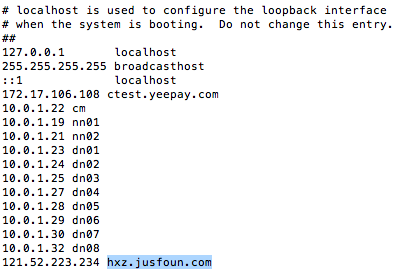
**问题描述：**将爬虫部署到服务器后，连接超时，爬虫获取不到网页内容，而在本地却可以。

**原因：**其一，可能是因为本地与服务器请求发送的ip地址不一样，很多网站都有很多个服务器，只是对外公开的域名一样。第二种原因可能是ip被服务器封掉，服务器察觉该ip请求异常，收到ip请求后，不回应。

**解决方法：**如果是第一种原因造成的，这时可以通过本地ping该域名，会获得一个ip,如下图所示：



然后在服务器上配置hosts，输入sudo vi /etc/hosts命令进入编辑hosts，然后配置域名与具体ip关联。219.141.171.1 www.court.gov.cn



如果是第二种原因造成的，那么只能更换ip.

1. **用Java代码编写爬虫需要注意事项**

**第一**：设置好请求参数

**第二**：一般请求都需要带上cookie，如果请求不成功，最好设置一下cookie参数，还有host参数以及User-Agent参数。

**第三**：在设置cookie的前提下还不成功，则设置一下referer参数

**第四**：如果这时还不成功，则需要检查请求参数，看看是否有中文参数，注意URL编码问题，有的需要URL编码有的则不需要。

**第五**：如果网站有iP限制，那么第一个突破点是设置x-forwarded-for参数，将这个参数设置为随机的，用jsoup请求设置，代码如下：

.header("x-forwarded-for", ((**int**)(Math.*random*()\*10000))%255+"."

+((**int**)(Math.*random*()\*10000))%255+"."

+((**int**)(Math.*random*()\*10000))%255

+"."+((**int**)(Math.*random*()\*10000))%255)

一般情况下，第一种方法都不可行，因此需要第二种方法，设置代理ip,如果爬虫是maven工程，可以使用jsoup代理包，需要配置pom文件。配置如下：

<dependency>

<groupId>fakepath</groupId>

<artifactId>jsoup</artifactId>

<version>1.8.1\_proxy</version>

</dependency>

然后用jsoup代理设置代理ip，代码如下：

.proxy(Type.***HTTP***,"124.202.182.182", 8118)//设置代理，前面是ip，后面的参数是端口号8118

**第六**：如果爬虫在本地可以跑，但是在服务器不行，并且服务器与本地访问的域名对应的ip都相同，那么则需要检查请求的url中是否包含端口号这类的字符，如果包含，则可能是服务器的防火墙屏蔽了这类的请求，这时解决方法要么修改防火墙配置，要么把请求URL中的端口号去掉，第一种方法可以治本，第二种方法可能获取不到数据。

比如：<http://sbcx.saic.gov.cn>：9080/tmois/wszhcx\_getLikeCondition.xhtml

这样的URL请求可能在服务器上无法访问到，这时可以把9080端口号去掉试试。

http://sbcx.saic.gov.cn/tmois/wszhcx\_getLikeCondition.xhtml