

# PageRank作业

---

ZY2006109-姬轶

## 一、Scala代码的简要说明讲解

```
// Create RDD (构造RDD函数)
val links = sc.parallelize(List(
  ("A",List("B","C","D")),
  ("B",List("A")),
  ("C",List("A","B")),
  ("D",List("B","C"))
)).partitionBy(new HashPartitioner(10)).persist()

// 每个页面的排序值初始化为 1.0
var ranks = links.mapValues(v => 1.0)

// For to Compute (十次循环计算更新ranks)
for(i <- 0 until 10){
  val contributions = links.join(ranks).flatMap{
    case(pageId, (links, rank)) =>links.map(link => (link, rank / links.size))
  }

  // 每个页面的排序值设为  $0.15 + 0.85 * \text{contributionsReceived}$ 
  ranks = contributions.reduceByKey((x,y) => x+y).mapValues(v => 0.15 + 0.85*v)
}

// 输出跑出的结果
ranks.collect().foreach(println)

// Save File (保存文件到"ranks"中)
ranks.saveAsTextFile("ranks")
```

## 二、Spark运行的截图

```
scala>


scala> ranks.collect().foreach(println)
(A,1.4729483816191942)
(B,1.151795159344013)
(C,0.8081470492728162)
(D,0.5671094097639748)

scala> // Save File

scala> ranks.saveAsTextFile("ranks")
```

### 三、ranks文件的截图

```
"      Sorted by      name
"      Sort sequence: [\\/$, \\<core\\%(\\.\\d\\+\\)\\=\\>,\\.h$,\\.c$,\\.cpp$,\\~\\=\\*$,*,\\.o$,\\.obj$,\\.info$,\\.sw
"      Quick Help: <F1>:help  -:go up dir  D:delete  R:rename  s:sort-by  x:special
"      =====
./
./
._SUCCESS.crc
.part-000000.crc
._SUCCESS
.part-000000
~
~
~
~
~
~
~
~
~
~
```

 part-00000 - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

[A,1.4729483816191942)

[B,1.151795159344013)

[C,0.8081470492728162)

[D,0.5671094097639748)