

unixbench 控制脚本分析

一、执行调用关系

main -> runTests -> runBenchmark -> runOnePass -> executeBenchmark ->commandBuffered

- 1. main:解析参数，设置copies数量默认为1和cpu核心数量（当核心数大于1时），对copies进行遍历
- 2. runTests: 确定copies不超过maxCopies数，对测试题进行遍历，最后调用indexResults函数计算总分
- 3. runBenchmark: 确定执行命令和需要执行的遍数，对遍数进行遍历，最后调用combinePassResults函数计算单项分数
- 4. runOnePass: 调用executeBenchmark，最后对所有copies结果求和。若copies数量大于1，则将每次copies中的分数相加，时间为所有copies时间的平均数。
- 5. executeBenchmark:调用commandBuffered进行命令的执行，进行copies次数的调用，最后调用readResults函数读取结果。
- 6. commandBuffered:创建子进程，在子进程中用perl的command命令执行测试项，通过无名字管道的方式进行通信。

二、单测试项结果分析（结果处理为Run中的combinePassResults函数）

- 1. 指定单测试项运行遍数（run pass num），通过变量testParams中的repeat参数指定：

命名	次数	说明
short	3	默认
long	10	最大
single	1	最小

- 2. 先根据每一遍的结果进行排序，去掉最差的1/3的结果，可通过log文件查看：dump score为舍去的结果，Count score为参与算分的结果。
- 3. 每一项原始结果，形如：COUNT|x|y|z 其中x为分数，y为时间单位，若y为0则x代表比率，z为标签符号。
- 4. 当y为时间单位时的计算公式：

$$score = e^{(\sum_{i=1}^{iterations} \log(\frac{count \cdot timebase}{time})) / iterations}$$

- score: 单项分数
- iterations:剩余有效结果的个数
- count:每个有效结果的值
- timebase:时间基本单位
- time:运行的总时间

当y为0时的计算公式：

$$score = e^{(\sum_{i=1}^{iterations} \log(count))/iterations}$$

三、总分结果分析（结果处理为Run中的indexResults函数）

1. index值计算公式:

$$index = \frac{score * 10}{baseline}$$

- score:算出的单项分数
- baseline:记录在pgms/index.base中的基准值

2. 总分计算公式:

$$SUM_SCORE = e^{(\sum_{i=1}^{tests_num} \log(\frac{score}{baseline})) / test_num} * 10$$

- test_num:一个类型中的测试项的个数，可见Run中的testCats变量。