

CPU-Pi 使用说明

一款开源的 CPU 双精度浮点计算性能测试工具

By 青阳龙野@kohgylw

1. 程序简介

本程序是一款 CPU 运算能力的工具，它通过对 π 值的近似值计算来测试运行设备 CPU 的双精度浮点运算性能。您可以通过设置线程模式来分别测试 CPU 的单任务性能和多任务性能，线程模式包括了从单线程到一百二十八线程共计九种档位供您选择。

本程序的测试过程：也就是计算 π 的近似值——是采用牛顿-莱布尼兹公式对反三角函数的展开：

$$\frac{\pi}{4} = \sum_{n=0}^n \frac{(-1)^n}{2n+1}$$

来实现的。

上面的公式中，自变量“n”即为测试程序需要进行的迭代次数——这一数值越大，则完成测试的运算压力也就越大，并且耗时也会越长，而最终得到的结果也越接近于 π 的真实值。

双精度运算能力作为 CPU/GPU 性能评价的重要指标之一，能够很大程度上反映出该产品的“快”与“慢”，具体评价标准请见“使用方法”一节。

2. 使用方法

2.1 安装 java 环境

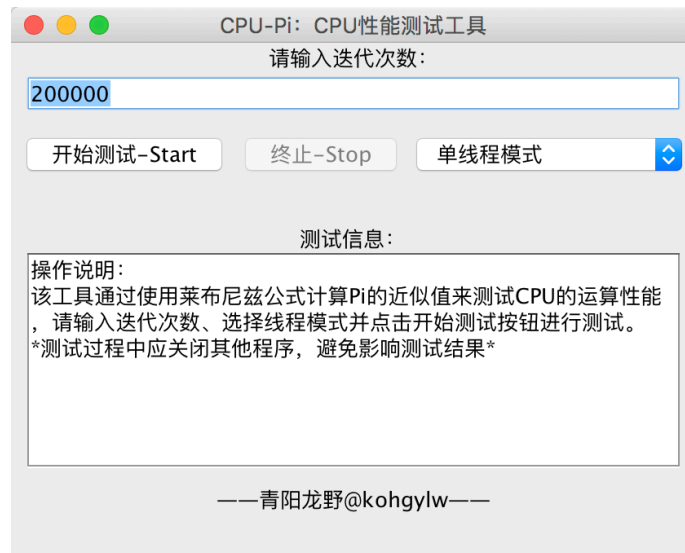
本程序使用 java 语言编写——这是一种广泛用于服务器和智能设备开发的语言，当然，也包括开发《我的世界》。想要运行本测试程序，必须先安装 java 环境。您可以去中文官网免费下载它：https://www.java.com/zh_CN/

安装 java 的过程非常简单——这比安装 VC 运行库等网游的必备环境简单得多，仅需一路下一步即可。

安装完 java 运行环境后，请运行“CPU-Pi.jar”来启动测试程序。

2.2 使用本程序

测试程序的主界面如下所示：



其中，位于上方的输入框用于设定测试压力——也就是执行运算的迭代次数。该数值越大，CPU 所要完成的运算量越大，得出的结果也就越接近 π 的真实值。该数值建议设定在 200000 及以上规模，如果运算规模太小则可能无法体现出低性能 CPU 与高性能 CPU 之间的差距。

当然，您也应该根据自己的 CPU 性能“量力而行”，超大规模的迭代次数除了会消耗更多的时间和电能外意义不大，该数值建议不要超过 1000000。

位于中部的按钮栏包括“开始测试”按钮、“终止”按钮和“线程模式”下拉菜单。

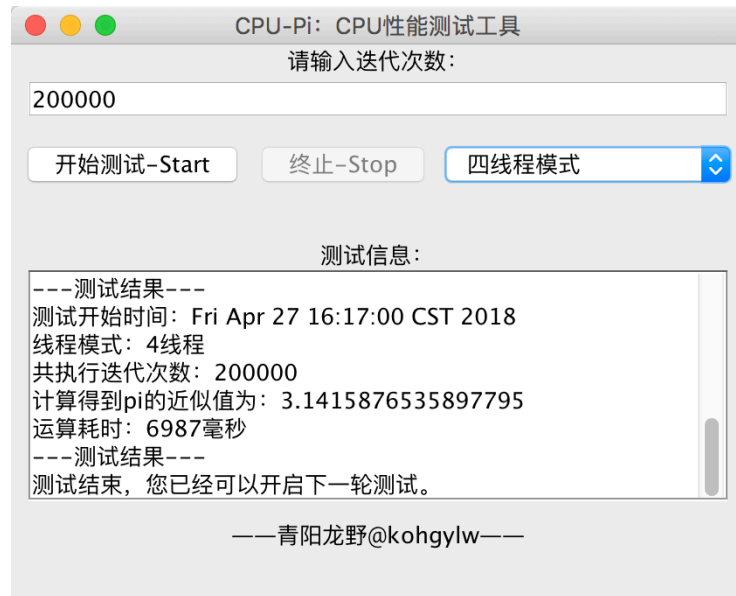
其中，**线程模式**下拉菜单是用于设定计算过程使用多少线程的。当您选择“单线程模式”的时候，主要测试的是 CPU 的单任务处理能力；如果您选择开启更多的线程完成运算，则主要测试 CPU 的多任务处理能力。注意，并不代表 4 核 4 线程的 CPU 使用 4 线程模式就是最佳运算模式——因为操作系统对于线程的调度过程并不完全遵循 CPU 支持的最大线程数来进行调度。事实上，对于一颗 4 核 4 线程的 CPU 而言，使用 6 线程模式的效率可能依旧要比使用 4 线程模式更高一些。您可以在测试中不断提高线程数来提高 CPU 的多任务压力，直到运算效率不再提高为止。

选择好线程模式并设定了测试压力（迭代次数）后，点击**开始测试**按钮执行测试过程。您可以从下方的**测试信息**文本框来监视实时的测试信息，这里会显示测试参数、测试进度以及最后的测试结果。

注意：测试过程中请先关闭其他应用程序以确保测试结果准确。

如果您觉得测试过程过慢需要终止测试，请点击**终止**按钮。

如果正常完成测试，则会在测试信息文本框中显示测试结果。例如下图所示：



测试结果包含如下几条信息：

开始时间——本次测试的开始时间记录，精确到秒。

线程模式——运算过程所开启的线程数。

共执行迭代次数——迭代任务将由所有线程平均分配，例如迭代次数为 200000 次，线程模式为 4 线程，则每个线程负责 50000 次迭代。

计算得到的 pi 的近似值——最终得到的计算结果。该值小数点后的位数越接近 π 的真实值，则运算效果越好，也意味着执行的迭代次数越大。该值对于测试结果而言毫无意义，仅用来表示测试程序的逻辑正确性。

运算耗时——该数值以毫秒为单位统计全部运算过程的耗时。这项数值直接体现 CPU 运算能力的，两台不同的设备在同一测试参数下所得到的耗时差异即为两台设备运算能力的差异。例如上图所示的运算结果即为：2016 款 Macbook Pro Intel 第 6 代 Core i5 双核 2.9Ghz 在 Mac OS X 操作系统、200000 次迭代、4 核模式下所获得的成绩。如果您的耗时比这个数值小，则代表您所使用的 CPU 比开发者的要好，反之则差。

您可以将该测试程序在不同的设备上运行以得到它们之间的性能差异，耗时越短代表其 CPU 运算性能越好。您可以将其分为单线程模式和多线程模式分别进行测试。

3.程序信息

本测试程序的**源代码**已经被打包在该 jar 包之中，如需查看请直接导入到您的 IDE 或将其解压后查看。

使用声明：当您使用该程序时意味着您已经接受这一条款——您可以免费使用该程序及其源代码，但不可将其用于任何形式的商业用途。

本程序作者为——青阳龙野/kohgylw/kohgylw@163.com。需要联系作者？请发送 E-mail 至 kohgylw@163.com，他随时欢迎您的来信。