

FEALPy 偏微分方程数值解程序设计与实现： 计算机基础

魏华祎, 王鹏祥

湘潭大学 • 数学与计算科学学院

July 5, 2020

目录

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 Git 版本控制

Outline

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 Git 版本控制

CPU

中央处理单元(CPU)主要由**运算器、控制器、寄存器**三部分组成,具有**控制时间、处理指令、执行操作、处理数据**四大作用。

现在常用的单颗 CPU 实际都集成多个中央处理单元,这些处理单元称为 CPU 的核心,这样的 CPU 称为**多核心处理器**,这些核心可以分别同时执行程序指令。

- 很多 CPU 的核心都支持**单指令多数据流 (SIMD)**,提供**指令级的并行功能**,即单个指令可以同时操作多组数据。
- 单个多核的 CPU 提供了**线程级的并行功能**,即多个核心同时执行多个指令流,同时处理多个数据流。
- 多个多核 CPU 一起可以提供了**进程级的并行功能**,即多个 CPU 同时运行多个进程,每个进程又可以有多个线程,同时处理更多个数据流。

CPU 核心

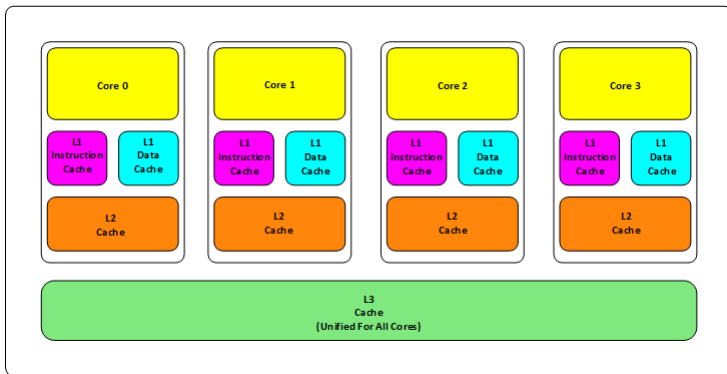


Figure: 多核 CPU 示意图。

CPU 主频

- CPU 在每秒内发出的时钟脉冲信号的个数, 即称为 CPU 的主频。
- 常用的单位是 GHZ, 如主频为 4.0 GHZ 的 CPU, 每秒可产生 40 亿个时钟脉冲信号。
- 通常主频越高的 CPU, 单位时间内执行程序指令的速度越快。

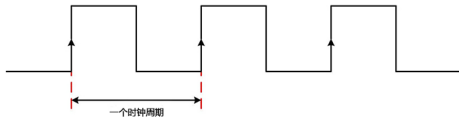


Figure: CPU 时钟信号。

Outline

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 Git 版本控制

内存

计算机内存是由很多微型开关组成的存储数据的硬件, 其中每个开关的开合状态分别代表 **0** 和 **1**。

0	1	0	1	0	0	1	1	0	1	0	0	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Figure: 计算机内存示意图。

内存

- 内存中的每个开关的状态称为一个**位 (bit)**, 是计算机中内存的最基本单位。
- 八个位组成一个**字节 (Byte)**。
- 为了获取内存中存储的指令和数据, 需要对内存中的每个**字节** 进行编号, 这个编号就是**物理内存地址**。

$$1KB = 1024B = 2^{10}Byte$$

$$1MB = 1024KB = 2^{20}Byte$$

$$1GB = 1024MB = 2^{30}Byte$$

$$1TB = 1024GB = 2^{40}Byte$$

- **计算机内存的结构是线性的**, 可以看成由很多可以存储 0 或 1 小格子组成的长条形结构。

内存

- 32 位的计算机, 用 32 位的二进制数表示内存的物理地址, 是大可支持 $2^{32}Byte = 2^2 \times 2^{30}Byte = 4GB$ 。
- 64 位的计算机, 用 64 位的二进制数表示内存的物理地址, 最大可支持 $2^{64}Byte = 2^{34}GB$ 。

Outline

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 Git 版本控制

简介

操作系统 (OS) 是我们操作计算机硬件的帮手, 而且很多时候我们还需要计算机同时处理多个任务, 每个任务又可能有多个执行流程, 但问题是计算机可利用的计算和存储等资源都是**有限**的。

所以操作系统的主要功能是: 在硬件资源有限的条件下, 实现多任务的并发运行, 尽量提高硬件资源的利用效率, 提供资源访问的冲突协调机制。

进程

进程：是 OS 进行计算机资源分配的最小单位。

- 是 OS 对正在运行的程序的一种抽象。
- 是应用程序的执行实例，每个进程是由私有的虚拟地址空间、代码、数据和其它各种系统资源组成。

线程

线程：是 CPU 核心调度和分配的基本单位。

- 每个进程至少有一个主执行线程, 主执行线程终止了, 进程也就随之终止。
- 除了进行主线程之外, 一个进程还可创建更多的线程。
- 一个进程的所有的线程共享访问进程拥有的资源, 但不能直接访问其它进程拥有的资源。
- 线程是用**逻辑内存地址**来访问进程的虚拟地址空间。
- 操作系统为每个进程内的线程提供资源的协调访问机制, 如互斥锁(Mutex)和信号量(Semaphore)。

总结

- 计算机的计算和存储资源是有限的。
- 计算机能表示的信息也是有限的。
- 程序设计者的任务是尽可能高效利用有限的计算和存储资源,完成更多的任务的处理。

Outline

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 Git 版本控制

简介

与图形化用户界面(GUI)一样, 命令行(CLI) 也是一种操作计算机系统的用户界面, 但命令行提供的是一种更高效的操作方式。

- 基本格式: “command [-options] parameter1 ...”
- 常用的 Linux 操作命令:
 - ☐ cd: 文件目录的遍遍历
 - ☐ ls: 显示文件夹下文件和子文件夹
 - ☐ touch: 创建文件
 - ☐ mkdir: 创建文件夹
 - ☐ rm: 删除文件和文件夹
 - ☐

Outline

- 1 CPU
- 2 内存
- 3 操作系统
- 4 命令行
- 5 **Git 版本控制**

简介

Git 是分布式版本控制系统, 它可以更好的对代码和文档进行版本管理, 并支持多人协作。

- 中文学习文档：<https://git-scm.com/book/zh/v2>。
- gitlab.com 和 github.com 是两个基于 git 的代码文档托管平台。
- 目前我们团队完全基于 git 来进行代码和文档的协作。