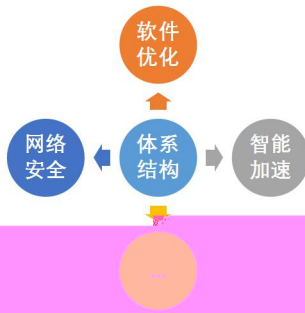
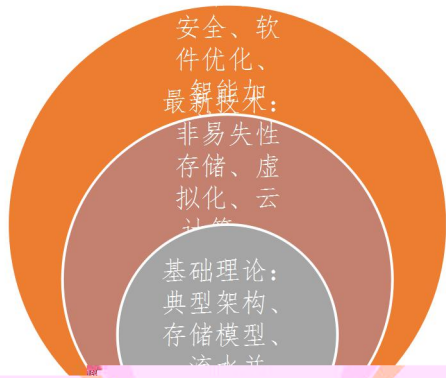


研究生精品课程简介





班级: 生班 学号: 20171010 姓名: 高基

题号	一	二	三	四	五	六	七	八	总分
成绩	10	8	19	22					89

** 注意: 所有题必须答在试卷上。

一、名词解释。

(1) NVM: 非易失性存储

(2) OoO: 乱序执行, 指 CPU 允许将多条指令不按程序序并发性地相对处理单元处理。

(3) IPC: CPU 每个时钟周期执行的指令数

(4) MFLOPS: 每秒百万次浮点运算

(5) CUDA: 统一计算设备架构, 使传统 GPU 通用

(6) SIMD: 单指令多数据

(7) VLIW: 超长指令字

(8) Cache Coherence: 缓存一致性, 当多个处理器在缓存一致性问题, 缓存中数据不一致时, 必须强制刷新, 增加延迟。

(9) Virtualization: 虚拟化, 消除不同系统接口知识

(10) Binary Translation: 二进制翻译, 将源程序翻

(10分)

处理器相关的...

...比, 专用...

...芯片、...

...的效果

...式计算机架构

...是计算机的...

...不应对...

...理系统...

... (exec...

...求计算

... (manta...

...ending

... (Saturati...

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

...指令

三、简答题。
1. 假设变量 A 和 B 的初始值均为 6，则可能是什么，并解释为什么。

P1		
A=1		
B=2		
A=1		
B=2		
A=1		
B=2		

答：① R1=1, P1 执行后 A=1, B=2
② R1=1, P1, P2, P3 有各自的 P1, P2, P3 执行顺序，并非按 P1, P2, P3 的顺序。P1 执行后 A=1, B=2, P2 执行后 A=1, B=1, P3 执行后 A=1, B=0。所以 R1=0。

2. 如下表给出了多种存储介质的参数对比，并从系统设计的角度来看如何在新的计算机系统中使用 PRAM?

表 1 多种存储介质的参数对比

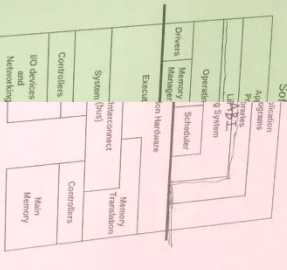
Medium	SRAM	DRAM	Flash/NonV	Flash/VR
Non-Volatile	No	No	Yes	Yes
Cell Size/pt	20-120	6-10	10	5
Read Time/ps	1-100	30	10	30
Write/erase Time/ps	1-100	15	100/100	100/100
Densities	10M	10M	105	105
Write Power	Low	Low	Var High	Var High
Other Power	Current	Current	Current	Current
Consumption	Low	Low	Low	Low
High Voltage Read/pt	No	Yes	Yes	Yes

PRAM 的工作原理：利用电介质玻璃在特定电压下产生不同极化状态，通过电介质玻璃的极化状态来存储数据。PRAM 具有非易失性，不需要刷新，且功耗低，集成度高。PRAM 的写入速度较慢，但读取速度很快。PRAM 的写入速度较慢，但读取速度很快。PRAM 的写入速度较慢，但读取速度很快。

(35) (25)
7. 请说明 PRAM (25) 的工作原

PRAM 的工作原理：利用电介质玻璃在特定电压下产生不同极化状态，通过电介质玻璃的极化状态来存储数据。PRAM 具有非易失性，不需要刷新，且功耗低，集成度高。PRAM 的写入速度较慢，但读取速度很快。PRAM 的写入速度较慢，但读取速度很快。

3. 请简要说明非易失性 PRAM (25) 的工作原理。非易失性 PRAM 利用电介质玻璃的极化状态来存储数据。PRAM 具有非易失性，不需要刷新，且功耗低，集成度高。PRAM 的写入速度较慢，但读取速度很快。PRAM 的写入速度较慢，但读取速度很快。



在 ABI (Application Binary Interface) 中，系统调用 (system calls) 是用户程序与操作系统内核之间的接口。系统调用允许用户程序请求操作系统提供的服务，如文件操作、进程控制、系统管理等。系统调用通常通过特定的指令或函数接口实现。

进程调度：操作系统通过调度算法来管理进程的执行顺序。常见的调度算法包括先来先服务 (FCFS)、短作业优先 (SJF)、时间片轮转 (RR) 等。进程调度的目的是提高系统的吞吐量和响应时间，确保公平性和效率。

内存管理：操作系统负责管理系统的物理内存和虚拟内存。内存管理包括内存分配、回收、交换、共享等操作。虚拟内存技术允许程序使用比实际物理内存更多的地址空间，提高了系统的灵活性和可扩展性。

该处理器采用基于总线监听 (Cache 一致 controller) 的 MSI 协议状态转换图, 并说系统的效率?



答: 该处理器采用 MSI 协议状态转换图



本处理器采用 Cache 系统同故障

- ① 增加 E (Exclusive) 状态, 当 Cache 中只有 P 本志, 增加了 Cache 一致性, 减少了通信阻塞。
- ② 增加 O (Owner) 状态, 需要管理数据

7. 请从存储顺序一致性 (Memory Consistency Model) 的角度说明, 如下 Java 代码并运行在多个处理器内核上运行时存在什么样的问题, 以及如何修复该问题。

```

class Foo {
    private Helper helper = null;
    public Helper getHelper() {
        if (helper == null)
            synchronized (this) {
                if (helper == null)
                    helper = new Helper();
            }
        return helper;
    }
}
    
```

答: 调用 `helper = new Helper()`; 使用 `new` 初始化对象时, 总共有两个步骤, 一是开辟空间, 此时 `helper` 已非空, 二是调用构造函数初始化。两个步骤是先后顺序, 所以会有一些时间和时间间隔, 如果别的处理器在步骤一和步骤二之间去读取 `helper`, 此时 `helper` 非空, 但未完全初始化, 就读取到未初始化的 `helper` 数据。

修复该问题, 可在 `getHelper` 方法中先声明变量, 然后通过 `new Helper()` 来初始化此局部变量, 然后将非空值赋给 `helper`, 从而避免了上述问题。

(24 分)

四、综合应用。
1、请列举自己研究领域的一个计算密集型的数据算法，说明使用 CPU、GPU 和硬件加速器哪种平台实现更合理一些，并解释为什么。

答：神经网络快速卷积算法；通过计算机在 GPU 上的数据并行，其得到硬件加速，也涉及数据传输信息。

计算密集型算法是逐点相互作用的乘、加、控制流等三个要素。在 GPU 上计算，以逐点作用为主，但 GPU 与 CPU 之间存在数据依赖，不能简单并行，而且不同层计算量不同，控制流不同。对于上述算法，可以采用 GPU 并行计算。通过设计合适的任务调度算法，将不同计算量任务分配给不同 CPU 核心，大计算量任务在 CPU 上，小计算量任务在 GPU 上，充分利用控制流在 CPU 上运行，数据流在 GPU 上运行。

2、如下为一指令序列，请分析这些指令之间的 RAW, WAR 和 RAR 相关性，并以此为例说明如何通过寄存器重命名实现多条指令并行执行。

(1) add	r3, r2, r3
(2) sub	r2, r1, r3
(3) mul	r1, r3, r1
(4) add	r2, r3, r1
(5) add	r2, r1, r3

答：RAW (Write-After-Write): (2) → (4), (4) → (5)

WAR (Write-After-Read): (1) → (2), (2) → (3), (1) → (4), (1) → (5), (3) → (4), (3) → (5)

RAW (Read-After-Write): (1) → (2), (1) → (3), (1) → (4), (1) → (5)

- 寄存器重命名
- (1) add r4, r2, r3
 - (2) sub r3, r1, r4
 - (3) mul r6, r3, r1
 - (4) add r7, r4, r6
 - (5) add r8, r6, r4

	r1	r2	r3
(1)	r1	r2	r3
(2)	r1	r2	r4
(3)	r1	r3	r4
(4)	r1	r3	r4
(5)	r1	r3	r4

指令重排执行

