北京师范大学2010～2011学年第二学期期末考试试卷（A卷）

课程名称： 计算机组成原理 任课教师姓名：

装 订 线

卷面总分： 100 分 考试时长： 100 分钟 考试类别：闭卷 开卷□ 其他□

院（系）： 专业： 年级： 姓名： 学 号：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 题号 | 一 | 二 | 三 | 四 | 五 | 总分 |
| 得分 | 40- | 20- | 12- | 14- | 14- | 100- = |

阅卷教师（签字）：

**第一部分：数据的表示和运算**

1. 将下面的32位值按照从小到大排列：

A. 0xF000-0000 (IEEE浮点数, IEEE float) B. 0xF000-0000 (补码, 2’s complement)

C. 0xF000-0000 (原码, sign-magnitude) D. 0xFFFF-FFFF (补码, 2’s complement)

E. 0xFFFF-FFFF (反码, 1’s complement) F. 0xF100-0000 (IEEE浮点数，IEEE float)

G. 0x7000-0000 (IEE浮点数，IEEE float) H. 0x7FFF-FFFF (补码, 2’s complement)

I. 0x8000-0000 (IEE浮点数，IEEE float)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 最小 | >> | >> | >> | >> | >> | >> | 最大 |
|  |  |  |  |  |  |  |  |

2. 给出16进制数0x811F00FE，按照不同的解释分别为：

A,B行： 请从右至左按4个数字为单位分开，如0000-1010或 207-0366；

C,D行： 请转换到十进制数值表示；

E行： 请以“|”符号分割各个指令域(假设为I型指令)。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | 二进制 |  | | | |
| B | 八进制 |  | | | |
| C | 四个8位无符号数 |  |  |  |  |
| D | 四个8位补码数 |  |  |  |  |
| E | MIPS指令 |  | | | |

3. 穿越题：

假设现在是1984年，而你是任天堂8位游戏机NES的开发者。现在你和你的团队想要在表示小数，它有所有IEEE754标准的属性 (包括隐藏的1，非规格化数，\_\_\_\_\_\_, 正负无穷)，当然它只有8位，而不是IEEE754单精度的\_\_\_\_\_位，所以表示形式、范围、精度是不同的。

最终你们决定将8位分成1( \_\_\_\_\_ ) + 3 (阶码) +4 (尾数)，即SEEEMMMM。而阶码的偏移为3 (IEEE754单精度中为\_\_\_\_\_\_ ), 而非规格化数对应的阶码为-2 (IEEE754单精度中为\_\_\_\_\_\_ )。

按照这个设计，这个数所能表示的最大数值为\_\_\_\_\_\_\_\_\_\_\_(正无穷除外)，对应的16进制表示为\_\_\_\_\_\_。最大（最接近0）的负数的值为\_\_\_\_\_\_\_\_\_\_\_\_，对应的16进制表示为\_\_\_\_\_。而-0.25，-4.0，和-0.125对应的16进制表示分别为\_\_\_\_\_\_、\_\_\_\_\_\_和\_\_\_\_\_\_。

最终，你们给这个数据类型起名为\_\_\_\_\_\_\_\_。

4、考研题：

浮点数加减运算过程一般包括对阶、尾数运算、规格化、舍入和判断溢出等步骤。假设浮点数的阶码和尾数均采用补码表示，且位数分别为5位和7位（均含两位符号位）。若有两个数X=27\*29/32; Y=25\*5/8，请填空：

(A) 假设阶码在前，X和Y的16进制表示分别为\_\_\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_\_\_。

(B) 浮点数加法运算X+Y的最终结果是\_\_\_\_\_\_\_\_。

**第二部分：存储系统**

1. 假设某计算机的存储系统有两级Cache。其中第一级Cache的命中率为90%，第二级Cache的(局部)命中率为40%，则两级Cache总的命中率为\_\_\_\_\_\_\_。

2. 假设MIPS机器的存储系统如下所示：

\* 1MB的物理地址空间 \* 4GB的虚拟地址空间 \* 4KB 页面大小

\* 16KB高速缓存 (Cache)

\* 8道组相联 (8-way, Set-Associative) \* 写透 (Write-through)

\* LRU替换策略 (Least-Recently-Used) \* 1KB块大小 (Cache Block Size)

\* 2-条目的快表(TLB), LRU替换策略

假设当前系统**仅**运行下面的代码段 (没有进程切换)：

*#define NUM\_INTS 8192*

*int \*A=(int\*)malloc(NUM\_INTS\*sizeof(int)); //返回地址 0x10-0000*

*for ( i = 0; i < NUM\_INTS; i+=128) A[i]=i;*

*for ( i = 0; i < NUM\_INTS; i+=128) total += A[i];*

请填空：

(A) 物理地址有\_\_\_\_\_位，可从Cache的角度，按照\_\_\_\_:\_\_\_\_\_:\_\_\_\_分割为\_\_\_\_:\_\_\_\_:\_\_\_\_。

(B) 虚拟地址有\_\_\_\_\_位，可从虚拟页的角度，按照\_\_\_\_ :\_\_\_\_\_分割为\_\_\_\_:\_\_\_\_。

(C) 每个缓存块可以存放\_\_\_\_个数组元素; 每个虚拟页可以存放\_\_\_\_个数组元素。

(D) 第一个for循环中，每\_\_\_\_次循环对应的地址对应一个缓存块(Cache Block)，循环\_\_\_次将耗光所有缓存容量，如存满A[0]到A[\_\_\_\_]；而第一个for循环结束后，缓存中存储的是A[\_\_\_\_]到A[\_\_\_\_]。

(E) 第二个for循环中，缓存、TLB和页表的命中率分别为：\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。

**第三部分：C => MIPS**

C:

|  |  |
| --- | --- |
| *typedef struct kid\_node{*  *int grade;*  *char \*name;*  *struct kid\_node \*next;*  *}kid\_t;* | *int grade\_sum (kid\_t) \*kid) {*  *if (kid==NULL)*  *return 0;*  *else*  *return kid->grade + grade\_sum(kid->next);* |

MIPS:

|  |  |
| --- | --- |
| grade\_sum: | #请补充注释 |
| beq \_\_\_\_\_, \_\_\_\_\_\_, NULL\_CASE | 对应if语句 |
|  | Move Stack Point |
|  | Store return address to stack |
|  |  |
|  | Prepare kid->next as argument for next call |
| jal grade\_sum |  |
| lw $a0, 0($sp) |  |
|  | Load kid->grade into $t0 |
|  | Restore return address |
|  |  |
|  | Prepare kid->grade + grade\_sum(kid->next); |
|  | return |
| NULL\_CASE: |  |
|  | Prepare 0 |
|  | Return |

指令提示：sw, lw, addu, addi, move, jr；寄存器提示：$a0, $v0, $t0, $ra, $sp

**第四部分：输入输出系统**

1、假设每个扇区512字节，磁盘转速为5400RPM，广告声称寻道时间为12ms，数据传输率为4MB/s,磁盘控制器开销为1ms，不考虑排队时间，则旋转延迟、传输时间、以及磁盘响应时间分别为\_\_\_\_\_\_\_、\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_。

2、假设处理器为1GHz时钟频率，每次轮检(Polling)花费400个时钟周期。假设鼠标需要每秒轮检30次，每秒需要花费\_\_\_\_\_\_个时钟周期，即\_\_\_\_\_\_\_\_\_%的CPU时间。软盘的数据传送单位为2字节，传送速度为50KB/秒，因此每秒需轮检软盘\_\_\_\_\_\_\_次，花费\_\_\_\_\_\_\_\_个时钟周期，即\_\_\_\_\_\_\_\_\_%的CPU时间。硬盘的数据传送单位为16字节，传送速度为16MB/秒，因此每秒需轮检磁盘\_\_\_\_\_\_次，花费\_\_\_\_\_%的CPU时间。如果改为中断方式，每次传送数据有500个时钟周期的延时。则在硬盘活跃(传输数据)期间，每秒硬盘中断次数为\_\_\_\_\_，花费\_\_\_\_\_\_%的CPU时间。假设硬盘有5%的时间是活跃的，中断方式下的硬盘数据传输处理花费\_\_\_\_\_\_\_%的CPU时间。

**第五部分：其他和综合**

1、根据下图给出的每条指令的频率和CPI(Clocks Per Instruction)填空。

|  |  |  |
| --- | --- | --- |
| 指令 | 频率(%) | CPI |
| ALU | 25 | 1 |
| Load | 35 | 3 |
| Store | 10 | 5 |
| Branch | 30 | 4 |

(A) 指令集总的CPI为\_\_\_\_\_。

(B) 如果Store指令的CPI为0，CPU将快\_\_\_\_\_倍。

(C) 如果能(且只能)使一条指令快一倍，应选择\_\_\_\_\_\_指令。

2、对每个左边的术语：

(1) 在括号内填写右边相关的描述标记，如无相关则画横线。

(2) 在填空处写出术语的全称(如果术语是简写)或英文(如果术语是中文)。

( ) LRU：\_\_\_\_\_\_\_ (A) 单位时间完成的工作

( ) 时间局部性：\_\_\_\_\_\_\_ (B) 完成单个任务的时间

( ) 空间局部性：\_\_\_\_\_\_\_ (C) 需要Cache较大才能利用这一点

( ) 并行：\_\_\_\_\_\_\_ (D) 所有的Cache都能利用了这一点

( ) AMAT：\_\_\_\_\_\_\_ (E) 一种用于选择替换对象的策略

( ) 超标量：\_\_\_\_\_\_\_ (F) 在慢速外设情况下较浪费CPU资源

( ) 流水线：\_\_\_\_\_\_\_ (G) 降低了Cache写命中时的开销

( ) 轮询：\_\_\_\_\_\_\_ (H) 存储系统的重要指标

( ) 吞吐率：\_\_\_\_\_\_\_

( ) 延迟：\_\_\_\_\_\_\_

( ) 写回：\_\_\_\_\_\_\_