北京大学信息科学技术学院考试试卷

考试科目：计算机系统导论 **姓名：**_____ **学号：**_____

考试时间： 2015 年 11 月 09 日 **小班教师：**_____

题号	一	二	三	四	五	**总分**
分数						
阅卷人						

北京大学考场纪律

1、考生进入考场后，按照监考老师安排隔位就座，将学生证放在桌面上。无学生证者不能参加考试；迟到超过 15 分钟不得入场。在考试开始 30 分钟后方可交卷出场。

2、除必要的文具和主考教师允许的工具书、参考书、计算器以外，其它所有物品（包括空白纸张、手机、或有存储、编程、查询功能的电子用品等）不得带入座位，已经带入考场的必须放在监考人员指定的位置。

3、考试使用的试题、答卷、草稿纸由监考人员统一发放，考试结束时收回，一律不准带出考场。若有试题印制问题请向监考教师提出，不得向其他考生询问。提前答完试卷，应举手示意请监考人员收卷后方可离开；交卷后不得在考场内逗留或在附近高声交谈。未交卷擅自离开考场，不得重新进入考场答卷。考试结束时间到，考生立即停止答卷，在座位上等待监考人员收卷清点后，方可离场。

4、考生要严格遵守考场规则，在规定时间内独立完成答卷。不准交头接耳，不准偷看、夹带、抄袭或者有意让他人抄袭答题内容，不准接传答案或者试卷等。凡有违纪作弊者，一经发现，当场取消其考试资格，并根据《北京大学本科考试工作与学术规范条例》及相关规定严肃处理。

5、考生须确认自己填写的个人信息真实、准确，并承担信息填写错误带来的一切责任与后果。

学校倡议所有考生以北京大学学生的荣誉与诚信答卷，共同维护北京大学的学术声誉。

以下为试题和答题纸，共　　页。

装订线内

不要答题

得分

第一题 单项选择题（每小题 1 分，共 20 分）

注：请将选择题答案填写在下表中

题号	1	2	3	4	5	6	7	8	9	10
答案										
题号	11	12	13	14	15	16	17	18	19	20
答案										

1. 给定一个实数，会因为该实数表示成单精度浮点数而发生误差。不考虑 NaN 和 Inf 的情况，该绝对误差的最大值为

   A. 2^{103}

   B. 2^{104}

   C. 2^{230}

   D. 2^{231}

   参考信息：单精度浮点数阶码 8 位，尾数 23 位

答案：a。计算过程：浮点数阶码 8 位，即最大指数为 127，尾数为 23 位，即最低位表示的数为$2^{-23} * 2^{127} = 2^{104}$。那么误差在刚好落在最低位的一半的时候最大，即$2^{103}$。其他选项为干扰项，如果求最大指数的时候忘记非规格化数或者对偶数舍入理解不深会算出 b，如果以为指数为非规格化数会算出 c 或 d。

2. 以下说法错误的是

   A. 负数加上负数结果可能为正数

   B. 正数加上正数结果可能为负数

   C. 用&和~可以表示所有的逻辑与或非操作

   D. 用&和|可以表示所有的逻辑与或非操作

答案：d。无法用&和|表示逻辑非

3. 在 32 位平台上，按 C90 标准以下语句中，结果为假的是

   A. `return INT_MIN < INT_MAX;`

   B. `return -2147483648 < 2147483647;`

   C. `int a = -2147483648; return a < 2147483647;`

   D. `return -2147483647-1 < 2147483647;`

   参考信息：

C90 的转换顺序：int -> long -> unsigned -> unsigned long

$2^{31} = 2147483648$

答案：b。因为 2147483648 超出了有符号数的表示范围，C90 的 C 语言会将其识别为 unsigned。注意在 32 位机器上 long 和 int 都是 32 位。

4. 关于浮点数，以下说法正确的是

A. 给定任意浮点数 a，b 和 x，如果 a>b 成立（求值为 1），则一定 a+x>b+x 成立

B. 不考虑结果为 NaN、Inf 或运算过程发生溢出的情况，高精度浮点数一定得到比低精度浮点数更精确或相同的结果

C. 不考虑输入为 NaN、Inf 的情况，高精度浮点数一定得到比低精度浮点数更精确或相同的结果

D. 给定任意浮点数 a，b 和 x，如果 a>b 不成立（求值为 0），则一定 a+x>b+x 不成立。

答案：d。如果不出现 NaN 和 Inf，浮点数是满足单调性的，这时 a 和 d 都成立。如果出现 NaN 和 Inf，那么任何时候比较运算的操作数有 NaN 和 Inf 就为 0，就只有 d 成立。关于 b 和 c，由于运算的偶然性，高精度不一定比低精度精确。

5. 已知下面的数据结构，假设在 Linux/IA32 下要求对齐，这个结构的总的大小是多少个字节？如果重新排列其中的字段，最少可以达到多少个字节？

```
struct {
  char a;
  double *b;
  double c;
  short d;
  long long e;
  short f;
};
```

A. 32, 28

B. 36, 32

C. 28, 26　　D. 26, 26

选择 A，Linux/IA32 要求 double 和 long long 在 4 字节边界上对齐，其中 a 占 1 字节，后面填充 1 字节，b 占 4 字节，填充 2 个字节，c 占 8 字节，d 占 2 字节，填充 2 字节，e 占 8 字节，f 占 2 字节，再填充 2 字节，共 32 字节；重新排列时，可以按字节数从大到小排列，最后，最少需要 8+8+4+2+2+1 再加 3 字节满足 4 字节对齐要求，共 28 个字节。

6. 下列寻址模式中，正确的是：

A. (%eax, , 4)

B. (%eax, %esp, 3)

C. 123

D. $1(%ebx, %ebp, 1)

选择 C. A 中不能省略 Ei，B 中比例因子错误，D 中偏移地址表示错误。

7. 假设存储器按“大端法”存储数据对象，已知如下的 C 语言数据结构：union { char c[2]; int i; }; 当 c 的值为 0x01, 0x23 时，i 的值为：

A. 0x0123

B. 0x2301

C. 0x01230000

D.不确定

选择 D. 因为 i 的两个高地址字节的值未知。

8. 假设某条 C 语言 switch 语句编译后产生了如下的汇编代码及跳转表：

```
    movl 8(%ebp), %eax                .L7:
    subl $48, %eax                    .long .L3
    cmpl $8, %eax                     .long .L2
    ja .L2                            .long .L2
    jmp *.L7(, %eax, 4)               .long .L5
                                      .long .L4
                                      .long .L5
                                      .long .L6
                                      .long .L2
                                      .long .L3
```

在源程序中，下面的哪些（个）标号出现过：

A. ‘2’, ‘7’

B. 1

C. ‘3’

D. 5

选择 C. 标号的取值范围为’0’ – ，’1’、’2’、’7’、’9’、…等没有出现。

9. x86 体系结构中，下面哪个说法是正确的？

A. leal 指令只能够用来计算内存地址

B. x86_64 机器可以使用栈来给函数传递参数

C．在一个函数内，改变任一寄存器的值之前必须先将其原始数据保存在栈内

D．判断两个寄存器中值大小关系，只需要 SF 和 ZF 两个条件码

答案：B

A.leal 指令做普通算术运算；C.caller saved 寄存器才需要；D.还需要 OF

10．下列的指令组中，哪一组指令只改变条件码，而不改变寄存器的值？

A．CMP, SUB

B．TEST, AND

C．CMP, TEST

D．LEAL, CMP

答案：C

SUB 和 AND 都同时会改变条件码和寄存器的值，LEAL 不改变改变条码。

11．下面有关指令系统设计的描述正确的是：

A．采用 CISC 指令比 RISC 指令代码更长。

B．采用 CISC 指令比 RISC 指令运行时间更短

C．采用 CISC 指令比 RISC 指令译码电路更加复杂

D．采用 CISC 指令比 RISC 指令的流水线吞吐更高

答案：C ，CISC 的比 RISC 代码复杂（如不定长），因此译码也更复杂，优点是代码更短。运行时间和吞吐都谁更高要依据实际应用。

12．一个功能模块包含组合逻辑和寄存器，组合逻辑单元的总延迟是 100ps，单个寄存器的延时是 20ps，该功能模块执行一次并保存执行结果，理论上能达到的最短延时和最大吞吐分别是多少？

A．20ns, 50GIPS

B．120ns，50GIPS

C．120ns，10GIPS

D．20ps,10GIPS

答案：B，主要考察延时和吞吐的区别，延时最小是 logic + 1 个 register，吞吐最高时是把 logic 划分成无穷多份， 每一级需要 20ps, 1000/20ps = 50GIPS

13．关于流水线技术的描述，错误的是：

A．流水线技术能够提高执行指令的吞吐率，但也同时增加单条指令的执行时间。

B．减少流水线的级数，能够减少数据冒险发生的几率。

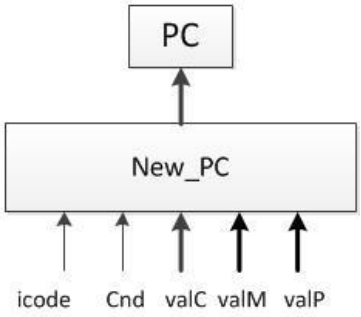
C．指令间数据相关引发的数据冒险，都可以通过 data forwarding 来解决。

D．现代处理器支持一个时钟内取指、执行多条指令，会增加控制冒险的开销。

答：C，load-use 冒险 不能通过 data forwarding 来解决

14．在 Y86 的 SEQ 实现中，PC（Program Counter，程序计数器）更新的逻辑结构如下图所示，请根据 HCL 描述为①②③④选择正确的数据来源。

```
Int new_pc = [
    # Call.
    Icode = ICALL : ① ;
    # Taken branch.
    Icode = IJXX && Cnd : ② ;
    # Completion of RET instruction.
    Icode = IRET : ③ ;
    # Default.
    1 : ④ ;
```



其中：Icode 为指令类型，Cnd 为条件是否成立，valC 表示指令中的常数值，valM 表示来自返回栈的数据，valP 表示 PC 自增。

A. valC, valM, valP, valP

B. valC, valC, valP, valP

C. valC, valC, valM, valP

D. valM, valC, valC, valP

答：C

15．下面关于存储器的说法，错误的是____

A．SDRAM 的速度比 FPM DRAM 快

B．SDRAM 的 RAS 和 CAS 请求共享相同的地址引脚

C．磁盘的寻道时间和旋转延迟大致在一个数量级

D．固态硬盘的随机读写性能基本相当

答案：D

说明：本题考查学生对内存、硬盘基本特性的掌握。

16．某磁盘的旋转速率为 7200 RPM，每条磁道平均有 400 扇区，则一个扇区的平均传送时间为____

A. 0.02 ms

B. 0.01 ms

C. 0.03 ms

D. 0.04 ms

答案：A

说明：60 / 7200 RPM × 1 / 400 sectors/track × 1000 ms/sec ≈ 0.02 ms

17. 某高速缓存 （E=2, B=4, S=16），地址宽度 14，当引用地址 0x9D28 处的 1 个字节时，tag 位应为：

A. 01110100

B. 001110000

C. 1110000

D. 10011100

答案：A

说明：0x9D28=10*01110100*101000

18. 关于高速缓存的说法正确的是 _____

A. 直写（write through）比写回（write back）在电路实现上更复杂

B. 固定的高速缓存大小，较大的块可提高时间局部性好的程序的命中率

C. 随着高速缓存组相联度的不断增大，失效率不断下降

D. 以上说法全不正确

答案：D

说明：本题考查学生对高速缓存基本性质和实现机制的理解。

19. 关于局部性（locality）的描述，不正确的是：

A. 循环通常具有很好的时间局部性

B. 循环通常具有很好的空间局部性

C. 数组通常具有很好的时间局部性

D. 数组通常具有很好的空间局部性

答案：C

数组本身只能体现出空间局部性

20. 以下哪些程序优化编译器总是可以自动进行？（假设 int i, int j, int A[N], int B[N], float m 都是局部变量，N 是一个整数型常量，int foo(int) 是一个函数）

	优化前	优化后

A.	for (j = 0 ; j < N ; j ++)
	m + = i*N*j;
		int temp = i*N;
		for (j= 0 ; j < N ; j ++)
		m + = temp * j;

B.	for (j = 0 ; j < N ; j ++)
	  B[i] *= A[j];
		int temp = B[i];
		for (j= 0 ; j < N ; j ++)
		  temp *= A[j];
		B[i] = temp;

C.	for (j = 0 ; j < N ; j ++)
	  m = (m + A[j]) + B[j];
		for (j = 0 ; j < N ; j ++)
		  m = m + (A[j] + B[j]);

D.	for (j = 0 ; j < foo(N) ; j ++)
	  m++;
		int temp = foo(N);
		for (j= 0 ; j < temp ; j ++)
		  m++;

答案：A

说明：考察 procedure, memory aliasing，和 floating 的精度问题

得分

第二题（20 分）

1.对于下面的每一个表达式，请选择以下选项中的一个或多个（即"不定项"），使得该表达式恒成立，如果没有满足条件的选项则选 E。

A. <　　B. >　　C. ==　　D. !=　　E. none

题目中出现的变量定义如下（浮点数保证不是 NaN 或者 Inf）：

```
int x, y;
unsigned ux = x;
double d;
```

1) 如果 x > 0，则 x + 1 ____ 0
2) 如果 x > y，则 ux _____ y
3) 如果 ((x << 31) >> 31) < 0，则 x & 1 _____ 0
4) 如果((unsigned char)x >> 1) < 64，则(char)x ____ 0
5) 如果 d < 0，则 d * 2 ____ 0
6) 如果 d < 0，则 d * d ____ 0
7) x ^ y ^ (~x) - y _____ y ^ x ^ (~y) - x
8) ((((!!ux)) << 31) >> 31) _____ ((((!!x) << 31) >> 31)

（前 6 题每题一分，漏选错选不得分；7、8 两题每题 2 分，漏选得 1 分;共 10 分）

2.考虑一种 12-bit 长的浮点数，此浮点数遵循 IEEE 浮点数格式，请回答下列问题。

注：浮点数的字段划分如下：

符号位（s）：1-bit；阶码字段（exp）：4-bit；小数字段（frac）：7-bit。

1) 请写出在下列区间中包含多少个用上面规则精确表示的浮点数（每空 2 分，共 4 分）

A: [1,2) _____

B: [2,3) _____

2) 请写出下面浮点数的二进制表示（每空 1 分，共 6 分）

数字	二进制表示
最小的正规格化数	
最大的非规格化数	
$17\frac{1}{16}$	

$-\frac{1}{8192}$	
$20\frac{3}{8}$	
$-\infty$	

答案：

1. 1)D 2)D 3)B,D 4) E 5)A,D

6)E 当两个数绝对值都很小时，相乘以后由于精度不够会变成 0

7)C 8)C

2. 1) A.128 B.64

2)

数字	二进制表示
最小的正规格化数	0 0001 0000000
最大的非规格化数	0 0000 1111111
$17\frac{1}{16}$	0 1011 0001000
$-\frac{1}{8192}$	1 0000 0000001
$10\frac{3}{8}$	0 1010 0100110
$-\infty$	1 1111 0000000

得分

第三题（20 分）

1.考虑下面的union的声明，回答后面的问题。

```
union ELE {
    struct {
        int x;
        int *p;
    }e1;
    struct {
        union ELE * next;
        int y;
    }e2;
};
```

(注：32位机器)

这个union总共大小为多少____字节(2分)

2.假设编译器为process的主体产生了如下了代码，请补充完整下面的过程：(6分)

（只有一个不需要任何强制类型转换且不违反任何类型限制的答案）

```
    movl 8(%ebp),%eax
    movl (%eax),%ecx
    movl 4(%ecx),%edx
    movl (%edx),%edx
    subl 4(%eax),%edx
    movl %edx,(%ecx)

void process(union ELE * up)
{
    up->_____=_____- _____;
}
```

3.请查看下文完成如下功能的汇编代码，定位错误语句并进行更正：（6 分）

给出 n(在%ebp+8 位置,n>=1),up(在%ebp+12 位置,ELE* 类型),假设以*up 为头元素(设*up 为第 0 个),由声明中的 next 连接形成了一个链表,请将第 n 个元素(假设链表足够长)的 x 的值放入%eax 中

X86 代码

```
    xorl %ecx,%ecx
    movl 8(%edx),%ebp
    movl 12(%ebp),%eax

    LOOP:
    movl (%eax),%eax
    add $1,%ecx
    test %ecx,%edx
    jne LOOP

    movl (%eax),%eax
```

Y86 代码

```
    mrmovl 8(%edx),%ebp
    mrmovl 12(%ebp),%eax
    irmovl $1,%ecx

    LOOP:
    mrmovl (%eax),%eax
    test %ecx,%edx
    jne LOOP

mrmovl (%eax),%eax
```

4.阅读下列代码，回答后面的问题

```
typedef struct {
    short x[A][B];
    int y;
```

```
}str1;

typedef struct {
    char array[B];
    int t;
    short s[B];
    int u;
}str2;

void setVal(str1 *p,str2 *q) {
    int v1=q->t;
    int v2=q->u;
    p->y=v1+v2;
}
```

(short 以 2 字节计算)

GCC 为 setVal 的主体产生下面的代码:

```
    movl 12(%ebp),%eax
    movl 28(%eax),%edx
    addl 8(%eax),%edx
    movl 8(%ebp),%eax
    movl %edx,44(%eax)
```

请直接写出A和B的值各是多少?(6分)

答案

1. 8
2. up->next->x=*(up->next->p)-(up->y)
3. (3个错, 每个错2分)

X86代码

```
xorl %ecx,%ecx
movl 8(%edx),%ebp     ||应为: movl 8(%ebp),%edx
movl 12(%ebp),%eax
```

```
LOOP:
movl (%eax),%eax
add $1,%ecx
test %ecx,%edx
jne LOOP

movl (%eax),%eax
```

Y86代码

```
mrmovl 8(%edx),%ebp   ||应为： mrmovl 8(%ebp),%edx
mrmovl 12(%ebp),%eax
irmovl $1,%ecx

LOOP:
mrmovl (%eax),%eax
test %ecx,%edx         ||应为：subl %ecx,%edx
jne LOOP

mrmovl (%eax),%eax
```

4.

A=3 B=7

得分

第四题（20 分）

请分析Y86 ISA中新加入的一条指令：NewJE，其格式如下。

NewJE	C	0	rA	rB	Dest

其功能为：如果R[rA]= R[rB]，则跳转到Dest继续执行，否则顺序执行。

1．若在教材所描述的SEQ处理器上执行这条指令，请按下表补全每个阶段的操作。需说明的信号可能会包括：icode, ifun, rA, rB, valA, valB, valC, valE, valP, Cnd；the register file R[], data memory M[], Program counter PC, condition codes CC。其中对存储器的引用必须标明字节数。如果在某一阶段没有任何操作，请填写none指明。

Stage	NewJE rA, rB, Dest
Fetch	icode:ifun ← M_1[PC] rA:rB ← M_1[PC+1] valC ← M_4[PC+2] valP ← PC+6
Decode	valA ← R[rA] valB ← R[rB]
Execute	valE ← valA – valB （注：也可以是valB – valA）
Memory	none
Write back	none
PC update	PC ← valE==0 ? valC : valP

（红字处每行 1 分，共 5 分）

2.若在教材所描述的PIPE处理器上执行NewJE指令，如果跳转条件不满足，一共

会错误执行__2__条指令。

为了减小错误预测的代价，现将教材所描述的PIPE处理器做如下改进：在Decode阶段增加一个比较器，用于判断（R[rA] = R[rB]）条件，比较器的输出信号为d_equal。如果相等，则d_equal = 1，反之 d_equal = 0。

此时，如果执行NewJE指令时跳转条件不满足，一共会错误执行__1__条指令。（每空 1 分，共 2 分）

3.在教材所描述的PIPE处理器上执行JXX指令时，发生转移预测错误的判断条件和各级流水线寄存器的控制信号如下所示：

Condition	Trigger
Mispredicted Branch	E_icode = IJXX & !e_Cnd

Condition	F	D	E	M	W
Mispredicted Branch	normal	bubble	bubble	normal	normal

在第（2）小题所述的改进后的处理器上执行 NewJE 指令，发生转移预测错误的判断条件和各级流水线寄存器的控制信号应如何设置？

Condition	Trigger
Mispredicted Branch	D_icode = INewJE & !d_equal

Condition	F	D	E	M	W
Mispredicted Branch	normal	bubble	normal	normal	normal

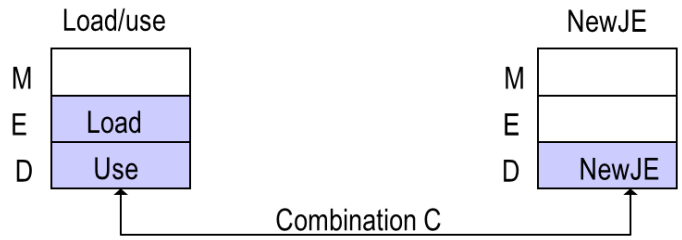
（每空 1 分，共 5 分）

4.在第（2）小题所述的改进后的处理器上执行如下代码，

```
0x000:    mrmovl 0(%eax), %edx
```

```
0x006:    NewJE %edx, %eax, t
0x00c:    irmovl $1, %eax    # Fall through
0x012:    nop
0x013:    nop
0x014:    nop
0x015:    halt
0x016: t:irmovl $3, %edx    # Target (Should not execute)
0x01c:    irmovl $4, %ecx    # Should not execute
0x022:    irmovl $5, %edx    # Should not execute
```

会发生 load-use 和 misprediction 组合的 hazard 情况，如下图所示



请问此时，各级流水线寄存器的控制信号应如何设置？

Condition	F	D	E	M	W
Combination C	stall	stall	bubble	normal	normal

（每空 1 分，共 3 分）

5.在教材 PIPE 处理器设计中，data memory 实际是高速缓存(cache)。假设在执行上述(4)中代码时，0x000 指令中的 0(%eax)地址中的数据不在 data memory 中，则 data memory 会将输出信号 m_datamiss 置为 1，直到数据从内存中取回到 data memory，再将 m_datamiss 置为 0。(m_datamiss 的默认值为 0)

这种情况的判断条件如下，请问各级流水线寄存器的控制信号应如何设置？

Condition	Trigger
Data Miss	M_icode in { IMRMOVL, IPOPL } && m_datamiss

Condition	F	D	E	M	W
data miss	stall	stall	stall	stall	bubble

（每空 1 分，共 5 分）

得分

第五题（20 分）

（每个填空和选择 1 分，表格每相邻两格 1 分）

1.仔细阅读下面的程序，根据条件回答下列各题（10 分）

- 地址宽度为 7，数组的起始地址为 0x1000000
- Block Size = 4 Byte，Set = 4，两路组相连.
- 替换算法位 LRU（最近最少使用）

```
#define LENGTH 8
void clear4x4 ( char array[LENGTH][LENGTH] )  {
    int row, col ;
    for ( col = 0 ; col < 4; col++ ) {
        for ( row = 0; row < 4 ; row++ ) {
            array[ row] [ col] = 0;
        }
    }
}
```

1） 以上程序执行会引起多少次失效?

4

2） 如果 LENGTH 改为 16，会引起多少次失效?

16

3） 如果 LENGTH 变为 17，与 2)相比，下面描述正确的是:A ，会引起 14 次失效。

A）16×16 比 17×17 产生更多的失效次数

B）16×16 和 17×17 产生的失效次数相同

C）16×16 比 17×17 产生更少的失效次数

4） 请画出 3）运行后 cache 中 set0 和 set1 的最终状态。

V	Tag	Data	V	Tag	Data
1	101	Array[1][0]~ Array[1][3]	1	100	Array[0][0]~ Array[0][3]
1	110	Array[2][0]~ Array[2][3]	1	111	Array[3][0]~ Array[3][3]

2.改变假设条件，回答下列各题（10 分）

- 地址宽度为 8，数组的起始地址为 0x10000000
- Cache 容量为 16 Byte，Block Size = 4 Byte，全相联 Cache
- 替换算法位 LRU（最近最少使用）

1） Tag 的位数为 6

2） 如果执行上述程序，当 LENGTH=8 时，会引起多少次失效？

4

3） 如果 LENGTH 改为 16，会引起多少次失效？

4

4） 如果 LENGTH 变为 17，与 3)相比，下面描述正确的是：C

A）16×16 比 17×17 产生更多的失效次数

B）16×16 和 17×17 产生的失效次数相同

C）16×16 比 17×17 产生更少的失效次数

5） 请画出 4）执行后 cache 的最终状态

V	Tag	Data
1	100000	Array[0][0]~ Array[0][3]
V	Tag	Data
1	100101	Array[1][0]~ Array[1][3]
V	Tag	Data
1	101001	Array[2][0]~ Array[2][3]
V	Tag	Data
1	101101	Array[3][0]~ Array[3][3]