

# 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： 2014 年 11 月 13 日 任课教师： \_\_\_\_\_

题号	一	二	三	四	五	六	七	八	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 15 页。

得分

### 第一题 单项选择题（每小题 2 分，共 32 分）

1、假设下列 unsigned 和 int 数均为 32 位，

```
unsigned x = 0x00000001;
```

```
int y = 0x80000000;
```

```
int z = 0x80000001;
```

以下表达式正确的是

A.  $(-1) < x$

B.  $(-y) > -1$

C.  $\sim y + y == -1$

D.  $(z << 4) > (z * 16)$

答：( )

2、下面说法正确的是：

A. 数 0 的反码表示是唯一的

B. 数 0 的补码表示不是唯一的

C. 1000, 1111, 1110, 1111, 1100, 0000, 0000, 0000 表示唯一的整数是 0x8FEFC000

D. 1000, 1111, 1110, 1111, 1100, 0000, 0000, 0000 如果是单精度浮点表示，则表示的是  $-(1.110111111)_2 * 2^{31-127}$

答：( )

3、下面表达式中为“真”的是：

A.  $(\text{unsigned}) -1 < -2$

B.  $2147483647 > (\text{int}) 2147483648U$

C.  $(0x80005942 \gg 4) == 0x09005942$

D.  $2147483647 + 1 != 2147483648$

答：( )

4、下列的指令组中，那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：( )

5、下列指令中，寻址方式不正确的是

- A. `MOVB %ah, 0x20(, %ecx, 8)`
- B. `LEAL (0xA, %eax), %ebx`
- C. `SUBB 0x1B, %bl`
- D. `INCL (%ebx, %eax)`

答：( )

6、有如下定义的结构，在 x86-64 下，下述结论中错误的是？

```
struct {  
    char c;  
    union {  
        char vc;  
        double value;  
        int vi;  
    } u;  
    int i;  
} sa;
```

- A. `sizeof(sa) == 24`
- B. `(&sa.i - &sa.u.vi) == 8`
- C. `(&sa.u.vc - &sa.c) == 8`
- D. 优化成员变量的顺序，可以做到“`sizeof(sa) == 16`”

答：( )

7、关于如何避免缓冲区溢出带来的程序风险，下述错误的做法为？

- A. 编程时定义大的缓冲区数组
- B. 编程时避免使用 `gets`，而采用 `fgets`
- C. 程序运行时随机化栈的偏移地址
- D. 在硬件级别引入不可执行代码段的机制

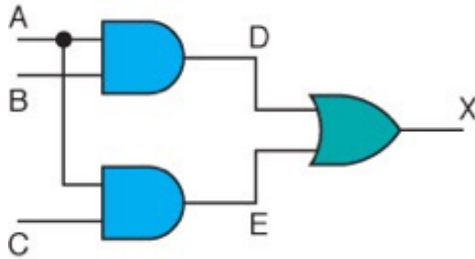
答：( )

8、对简单的 `switch` 语句常采用跳转表的方式实现，在 x86-64 系统中，下述最有可能正确的 `switch` 分支跳转汇编指令为哪个？

- A. `jmp .L3(, %eax, 4)`
- B. `jmp .L3(, %eax, 8)`
- C. `jmp *.L3(, %eax, 4)`
- D. `jmp *.L3(, %eax, 8)`

答：( )

9、对应下述组合电路的正确 HCL 表达式为



- A. `Bool X = (A || B) && (A || C)`
- B. `Bool X = A || (B && C)`
- C. `Bool X = A && (B || C)`
- D. `Bool X = A || B || C`

答：( )

10、若处理器实现了三级流水线，每一级流水线实际需要的运行时间分别为 2ns、2ns 和 1ns，则此处理器不停顿地执行完毕 10 条指令需要的时间为：

- A. 21ns
- B. 22ns
- C. 23ns
- D. 24ns

答：( )

11、关于 RISC 和 CISC 的描述，正确的是：

- A. CISC 指令系统的指令编码可以很短，例如最短的指令可能只有一个字节，因此 CISC 的取指部件设计会比 RISC 更为简单。
- B. CISC 指令系统中的指令数目较多，因此程序代码通常会比较长；而 RISC 指令系统中通常指令数目较少，因此程序代码通常会比较短。
- C. CISC 指令系统支持的寻址方式较多，RISC 指令系统支持的寻址方式较少，因此用 CISC 在程序中实现访存的功能更容易。
- D. CISC 机器中的寄存器数目较少，函数参数必须通过栈来进行传递；RISC 机器中的寄存器数目较多，只需要通过寄存器来传递参数。

答：( )

12、关于流水线技术的描述，正确的是：

- A. 指令间数据相关引发的数据冒险，一定可以通过暂停流水线来解决。
- B. 流水线技术不仅能够提高执行指令的吞吐率，还能减少单条指令的执行时间。
- C. 增加流水线的级数，一定能获得性能上的提升。
- D. 流水级划分应尽量均衡，不均衡的流水线会增加控制冒险。

答：( )

13、下面关于程序性能的说法中，哪种是正确的？

- A. 处理器内只要有多个功能部件空闲，就能实现指令并行，从而提高程序性能。
- B. 同一个任务采用时间复杂度为  $O(\log N)$  算法一定比采用复杂度为  $O(N)$  算法的执行时间短
- C. 转移预测总是能带来好处，不会产生额外代价，对提高程序性能有帮助。
- D. 增大循环展开 (loop unrolling) 的级数，有可能降低程序的性能 (即增加执行时间)

答：( )

14、仅考虑以下代码，哪些程序优化总是被编译器自动进行？（假设 `int i, int j, int A[N], int B[N], int m, int *p` 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. <pre>for (j = 0 ; j &lt; N ; j ++)   B[i] *= A[j];</pre>	<pre>int temp = B[i]; for (j= 0 ; j &lt; N ; j ++)   temp *= A[j]; B[i] = temp;</pre>
B. <pre>for (j = 0 ; j &lt; N ; j ++)   m + = i*N*j;</pre>	<pre>int temp = i*N; for (j= 0 ; j &lt; N ; j ++)   m + = temp * j;</pre>
C. <pre>i = foo(N); j = foo(N); if (*p != 0)   m = j ;</pre>	<pre>j = foo(N); if (*p != 0)   m = j ;</pre>
D. <pre>for (j = 0 ; j &lt; foo(N) ; j ++)   m ++;</pre>	<pre>int temp = foo(N); for (j= 0 ; j &lt; temp ; j ++)   m ++;</pre>

答：( )

15、以下关于存储结构的讨论，那个是正确的

- A. 增加额外一级存储，数据存取的延时一定不会下降
- B. 增加存储的容量，数据存取的延时一定不会下降
- C. 增加额外一级存储，数据存取的延时一定不会增加
- D. 以上选项都不正确

答：( )

16、关于 cache 的 miss rate，下面那种说法是错误的。

- A. 保持 E 和 B 不变，增大 S，miss rate 一定不会增加
- B. 保持总容量和 B 不变，提高 E，miss rate 一定不会增加
- C. 保持总容量和 E 不变，提高 B，miss rate 一定不会增加
- D. 如果不采用“LRU”，使用“随机替换策略”，miss rate 可能会降低

答：( )

得分

## 第二题 (10 分)

1) 假设下列 unsigned 和 int 数均为 5 位 (有符号整型用补码运算表示), 在下表中填入正确答案 (每空 1 分, 共 6 分)

```
int y = - 7;
unsigned z = y;
```

	Decimal Representation	Binary Representation
$z$		
$y - z$		
TMin		

2) 请按 IEEE 浮点标准的单精度浮点数表示下表中的数值, 首先写出形如  $(-1)^s \times M \times 2^E$  的表达式, 然后给出十六进制的表示。(每格 1 分, 共 4 分)

注: 单精度浮点数的字段划分如下:

符号位 (s): 1-bit; 阶码字段 (exp): 8-bit; 小数字段 (frac): 23-bit; 偏置值 (bias): 127。

Value	$(-1)^s \times M \times 2^E, 1 \leq M < 2$	Hex representation
0.375		
-12.5		

得分

### 第三题（10 分）

阅读以下代码。假设代码运行在 IA32 的计算机上，字长为 4，请给出各个变量在内存中的十六进制字节表示（地址从小到大）。

注意 tiny\_float 是一种 8 位的浮点数，1 个符号位，4 个指数位，3 个尾数位。

```
int main()
{
    int a = 0x15213;
    unsigned char b = ((char)-5);
    tiny_float c = 19;
    float d;
    if (b < 4)
        d = -0.96875;
    else
        d = 769;
}
```

答：



得分

#### 第四题 (10 分)

一个函数如下，其中部分代码被隐去，请通过gdb调试信息补全代码（4分）。

```
int f(int n, int m) {
    if (m > 0) {
        if (_____) {
            int r = _____;
            return _____;
        }
        else if (_____) {
            return 1;
        }
    }
    return 0;
}
```

如下是通过“gcc -g -O2”命令编译后，在gdb中通过“disas f”命令得到的反汇编代码，其中有两个汇编指令不全，请补全这两条汇编指令（2分）。

```
0x00000000004004e0 <f+0>:      mov     %rbx,-0x10(%rsp)
0x00000000004004e5 <f+5>:      mov     _____
0x00000000004004ea <f+10>:     xor     %eax,%eax
0x00000000004004ec <f+12>:     sub     $0x10,%rsp
0x00000000004004f0 <f+16>:     test    %esi,%esi
0x00000000004004f2 <f+18>:     mov     %edi,%ebp
0x00000000004004f4 <f+20>:     mov     %esi,%ebx
0x00000000004004f6 <f+22>:     jle     0x400513 <f+51>
0x00000000004004f8 <f+24>:     cmp     $0x1,%edi
0x00000000004004fb <f+27>:     jle     0x400521 <f+65>
0x00000000004004fd <f+29>:     lea     -0x1(%rbp),%edi
0x0000000000400500 <f+32>:     callq  0x4004e0 <f>
0x0000000000400505 <f+37>:     lea     -0x1(%rax,%rbx,1),%edx
0x0000000000400509 <f+41>:     mov     %edx,%eax
0x000000000040050b <f+43>:     sar     $0x1f,%edx
0x000000000040050e <f+46>:     idiv    %ebp
0x0000000000400510 <f+48>:     lea     0x1(%rdx),%eax
```

```

0x00000000000400513 <f+51>:      mov      _____
0x00000000000400517 <f+55>:      mov      0x8(%rsp),%rbp
0x0000000000040051c <f+60>:      add      $0x10,%rsp
0x00000000000400520 <f+64>:      retq
0x00000000000400521 <f+65>:      sete    %al
0x00000000000400524 <f+68>:      movzbl  %al,%eax
0x00000000000400527 <f+71>:      jmp     0x400513 <f+51>

```

已知在调用函数f(4, 3)时，我们在函数f中指令retq处设置了断点，下面列出的是程序在第一次运行到断点处暂停时时，相关通用寄存器的值。请根据你对函数及其汇编代码的理解，填写当前栈中的内容。如果某些内存位置处内容不确定，请填写x。（4分）

```

rax      0x1
rbx      0x3
rcx      0x3
rdx      0x309c552970
rsi      0x3
rdi      0x1
rbp      0x2
rsp      0x7fffffff340
rip      0x400520

```

0x7fffffff38c	
0x7fffffff388	
0x7fffffff384	
0x7fffffff380	
0x7fffffff37c	
0x7fffffff378	
0x7fffffff374	
0x7fffffff370	
0x7fffffff36c	
0x7fffffff368	
0x7fffffff364	
0x7fffffff360	
0x7fffffff35c	
0x7fffffff358	
0x7fffffff354	
0x7fffffff350	
0x7fffffff34c	
0x7fffffff348	
0x7fffffff344	
0x7fffffff340	
0x7fffffff33c	
0x7fffffff338	
0x7fffffff334	
0x7fffffff330	
0x7fffffff32c	
0x7fffffff328	
0x7fffffff324	
0x7fffffff320	

得分

### 第五题（8 分）

阅读下面的汇编代码，根据汇编代码填写 C 代码中缺失的部分，然后描述该程序的功能。

pushl %ebp	int fun(_____x) {
movl %esp,%ebp	int bit_sum = 0;
movl \$0x0, %ecx	while (_____) {
cmpl \$0x0, 8(%ebp)	_____;
jle .L1	_____;
.L2	}
movl \$0x0, %edx	if (_____)
movl 8(%ebp), %eax	return 1;
divl \$0x0a	else
addl %edx, %ecx	return 0;
movl %eax, 8(%ebp)	
cmpl \$0x0, 8(%ebp)	
jg .L2	
.L1	
movl 0x0, %edx	
movl %ecx, %eax	}
divl 0x3	
cmpl 0x0, %edx	
jne .L3	
movl 0x1, %eax	
jmp .L4	
.L3	
movl 0x0, %eax	
.L4	

答：

得分

**第六题（10 分）**

请分析Y86 ISA中新加入的一条指令：caddXX，条件加法。其功能可以参考add和cmovXX两条指令。

caddXX	C	fn	rA	rB
--------	---	----	----	----

若在教材所描述的SEQ处理器上执行这条指令，请按下表填写每个阶段进行的操作。需说明的信号包括：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	caddXX rA, rB
Fetch	
Decode	
Execute	
Memory	
Write back	
PC update	

得分

### 第七题（10 分）

如下是使用 C 语言描述的链表结构的声明，链表的结尾使用空指针来表示。同时使用函数 `int length (List *p)` 来计算链表的长度。为简化起见，假设该链表是非循环的。

```
typedef struct LIST {
    struct LIST *next;
    int data;
} List;
```

- 1) 函数 `count_pos1` 用来计算链表中 `data` 为正数的元素个数，并将结果存放在地址 `k`。以下的程序可能存在问题导致效率很低或程序出错，请指出并修改。

（4 分）

```
void count_pos1 (List *p, int *k) {
    int i;
    for (i = 0; i < length(p); i++) {
        if (p->data > 0)
            *k++;
        p = p->next;
    }
}
```

- 2) 为提高程序性能, 可以考虑删除变量 `i` 以消除函数调用。请修改上述程序达到该目的。(2 分)

- 3) 上述程序内层循环的汇编片段如下所示。假设该链表不为空且大部分数据都为正数, 转移预测全部正确, 设计中有足够多的部件来实现指令并行。其中访存操作全部 `cache` 命中, 时延为 3 cycle, 其他指令时延为 1cycle。请计算以下程序的 CPE 下限, 并给出文字说明。(4 分)

```
.L1:
    movl    4(%eax), %ecx
    testl   %ecx, %ecx
    jle     .L2
    incl    %edx

.L2:
    movl    (%eax), %eax
    testl   %eax, %eax
    jne     .L1
```

得分

### 第八题（10 分）

假设存在一个能够存储四个 Block 的 Cache，每一个 Block 的长度为 2Byte。假设内存空间大小共是 16Byte，即内存空间地址长度一共是 4bit，可访问地址为 (0~15)，数据访问地址序列如下所示，访问数据单位是 Byte，默认替换策略是 LRU。

**2    3    10    9    6    8**

1) 如果 Cache 的结构是下图所示 (S=2, E=2)，请在下图空白处填入访问上述六次数据访问后 Cache 的状态。注：用 [0-1] 表示地址 0 至 1 上对应的数据（4 分）

	V	TAG	Block	V	TAG	Block
set0						
set1						

2) 这六次数据访问一共产生了多少次 Miss \_\_\_\_\_（2 分）

3) 如果 Cache 的替换策略改成 MRU（即，最近使用的数据被替换出去），请在下图空白处填入访问上述六次数据访问后 Cache 的状态（2 分）。

	V	TAG	Block	V	TAG	Block
set0						
set1						

4) 这六次数据访问一共产生了多少次 Miss \_\_\_\_\_（2 分）