主管领导审核签字

哈尔滨工业大学 2022 年春季学期

计算机系统（A）　试　题

题号	一	二	三	四	五	六	七	总分
得分								
阅卷人								

片纸鉴心　诚信不败

授课教师　　姓名　　学号　　班级

一、 单项选择题（每小题 1 分，共 10 分）

1. 在 C 语言程序的链接阶段，链接器生成的文件不可能是（　　）。
   A. 后缀是.o 的目标文件
   B. 后缀是.so 的共享库文件
   C. 可执行程序
   D. ELF 格式的文件
2. 关于汇编指令 movl $0x1234, 8(%rax,%rbx,8)，错误的说法是(　　)。
   A. 目的操作数是内存操作数
   B. 内存操作数的地址是 8+(%rax+%rbx×8)
   C. 操作数宽度是 4 字节
   D. 可以改写成 mov $0x1234, 8(%rax, %rbx,8)，效果一样
3. 条件跳转指令 JC 是依据（　　）做是否跳转的判断。
   A.CF　　B. OF　　C. SF　　D. ZF
4. 在 C 语言程序中，按行优先顺序访问数组元素往往有较高效率的原因是(　　)。
   A. 在 C 语言程序中，数组是按行优先顺序在内存中存放的
   B. 这样的存储访问有比较好的空间局部性
   C. 能提高缓存的命中率
   D. 以上都对
5. 内核保存的、用于实现进程调度的进程上下文信息是（　　）。
   A. 全局变量值　B. 局部变量值　C. 寄存器　D. 磁盘文件
6. 一个进程用 fork 函数创建子进程成功时，fork 函数在该进程(父进程)中的返回值为(　　)。
   A. 0　　B. 子进程 ID　　C. 子进程组 ID　　D. 父进程的子进程数量
7. 使用函数 execve 启动一个程序时，错误的叙述是（　　）。
   A. 要删除原进程的页表和 vm_area_struct 链表
   B. 创建新的页表和 vm_area_struct 链表
   C. 将.bss 设定为请求二进制零、映射到匿名文件且初始长度为 0 的区域；
   D. 将栈设定为请求二进制零的、映射到匿名文件且初始长度为 0 区域也；
8. 在 X86-64 Linux 平台下，关于 64 位 C 语言程序中函数参数传递的说法，错误的是(　　)
   A. 全部采用寄存器传递参数

B. 有专门的寄存器用于浮点型参数值的传递
C. 参数的数量较多时，也采用栈来传递
D. 采用寄存器传参，能提高程序速度

9. 在 X86-64 Linux 平台下的 64 位程序中，关于 C 语言函数的说法错误的是（ ）
A. 非 static 局部变量可以在栈中
B. 非 static 局部变量可能在寄存器中
C. static 类型的局部变量也可能在栈中
D. 参数可能在栈中

10. 可以用函数( )，通过复制文件描述符指针和打开文件表节点引用计数的方式，实现输出重定向。
A. copy B. write C. mov D. dup2

二、填空题（每空 1 分，共 10 分）

11. Linux 下，将源程序 hello.c 编译生成可执行文件 hello 的指令是__①__，将可执行文件 hello 反汇编结果输出到指定的文本文件 1.txt 的命令是：__②__。

12. Linux 下查看作业的指令是_____。

13. C 语言程序定义了结构体 struct noname{ int n; char c; short k; char *p;};若该程序编译成 64 位可执行程序，则 sizeof(noname)的值是_____。

14. C 语言程序中的局部变量可能在__①__中或__②__中，也可能被编译器优化掉而在可执行文件中消失。

15. 在计算机的存储体系中，速度最快的是_____

16. 在 X86-64 Linux 平台下，C 语言函数调用时，整数值类型的参数用__①__、__②__、%rdx、%rcx 、%r8、%r9 等最多 6 个寄存器来传递，整数值类型的函数返回值则由寄存器__③__传递。

三、判断对错（每小题 1 分，共 10 分，在题前打√或 X 符号表示对、错）

17. （ ）小尾顺序也称小端序，是指整数值在内存中存放时，将低位数值存在低地址的内存字节中。

18. （ ）在 Y86-64 的流水线实现中，采用了阶段寄存器来保存上一条指令的运行结果，从而允许阶段部件能尽快运行下一条指令。

19. （ ）链接时，static 修饰的函数会被当成局部符号，不能被其他模块引用。

20. （ ）在 Intel CPU 中，有专门保存当前正在运行进程的页表首地址的寄存器。

21. （ ）C 语言中数值从 float 转换成 int 后，数值不会溢出。

22. （ ）向偶数舍入是为了产生统计上没有偏差的结果。

23. （ ）C 语言程序中，在用 malloc 函数申请内存空间后、使用申请的内存前，程序占用的物理内存立即会有相应数量的增加。

24. （ ）在 shell 中，当用户按下快捷键 ctrl+c 后，shell 应该将 SIGINT 信号发送给前台进程组的所有进程。

25. 多个运行中的进程可以通过私有的写时复制方式来共享物理内存中的共享库，节省计算机的内存开销。

26. （ ）C 语言的函数 printf 并不是异步信号安全的，在信号处理程序中使用该函数有潜在的安全风险。

四、系统分析题（30 分）

27. 阅读如下函数 myproc 的汇编代码，在横线上说明对应语句的功能（5 分）

```
myproc:
    movl    $0, %edx        ①
    movl    $0, %eax
    jmp  .L2
.L3:
    addq(%rdi,%rdx,8), %rax     ②
    addq$1, %rdx                ③
.L2:
    cmpq    %rsi, %rdx          ④
    jl   .L3                    ⑤
    ret
```

28. 写出 27 题函数 myproc 的 C 语言源程序（5 分）

29. 顺序结构 Y86 CPU 的抽象结构如图 1 所示，该 CPU 每个周期执行一条指令，CC 是三位二进制状态码，从高到低每位分别对应 ZF、SF、OF。%rbx、%rdx、CC 的初值为都是 100，按图 2 顺序执行指令序列。

（1） 在周期 3 结束前一刻即图中时刻①时，CC、PC、%rbx、%rdx 的数值分别是多少？组合逻辑是哪条指令的状态或结果？(5 分).

（2） 图 1 中的组合逻辑、时序逻辑分别包括哪些部件？组合逻辑、时序逻辑的特点或不同之处是什么？(5 分)

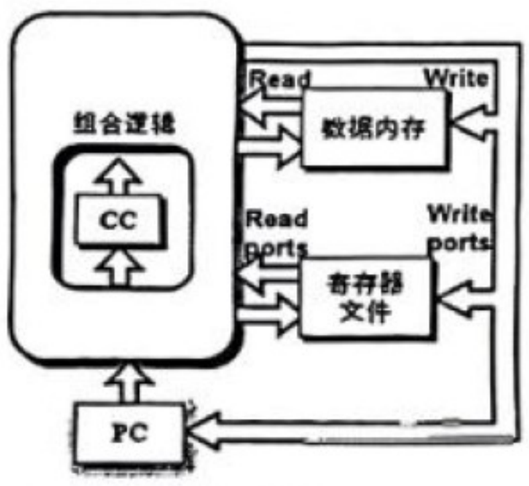

图 1 顺序结构 Y86 CPU 示意图

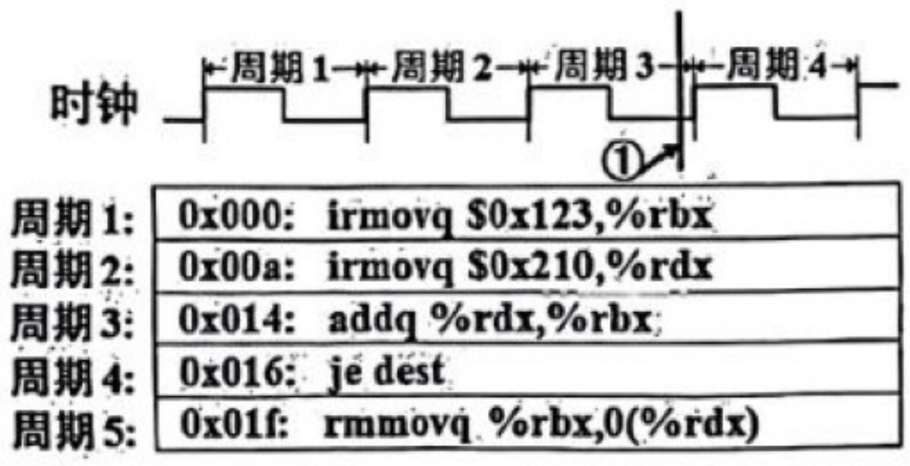

图 2 CPU 执行的指令序列

30. 有如下 C 语言程序，请画出程序的进程图，并给出可能的输出(10 分)。

```
int main()
{
    printf("L0\n");
    if (fork() != 0) {
        printf("L1\n");
        if (fork() != 0) {
            printf("L2\n");
    }
    }
    printf("Bye\n");
    return 0;
}
```

（1）进程图(5 分）
（2）可能的输出（5 分）

五、综合设计题（10 分）

31. 若一计算机系统的情况如下：

- 内存是按字节寻址，字长为 1 字节（而非 4 字节）；
- 虚拟地址长度：25 位/bits，虚拟页面大小是 2*1024 字节；
- 使用一级页表；
- 物理页号 PPN 的长度是 8 位/bits；
- L1 d-cache 是物理寻址、直接映射（每组一行），行大小 8 字节，共 8 个组，当前数值如表 1 所示。

表 1 L1 d-cache 的数值

索引	标记位	有效位	块 0	块 1	块 2	块 3	块 4	块 5	块 6	块 7
0	35E	1	42	A0	75	50	42	0	05	50
1	27B	1	08	E3	00	A7	13	00	8B	52
2	C54	1	3F	75	AB	11	25	78	9A	00
4	A32	1	97	3A	91	D3	3F	12	86	22
5	C30	1	30	62	15	4C	48	A1	12	5C
6	B26	1	01	25	3E	62	1F	C4	85	12
7	01A	1	98	3A	12	D39	3F	3C	4D	5E

回答以下问题：

（1） 计算页表条目 PTE 的总数量 、物理地址的位数、虚拟页号 VPN 的位数。
（2） 在 L1 d-cache 寻址中，块偏移、组索引、标记位的位长分别是多少？
（3） 在 CPU 从物理地址 0x31511 读取 1 字节数据，在访问缓存时，对应的组索引、标记位的数值、能否命中缓存、若能命中数值是多少？

六、简答题（每小题 6 分，共 30 分）

32. 简述链接器在用目标文件生成可执行文件时如何处理符号引用/解析的。

33. 以 gets 读入字符串为例，简述缓冲区溢出攻击的原理以及防范方法。

34. 程序速度性能优化的常用方法，及其能有效提高程序速度的原理。

35. CPU 执行从内存读数据的指令，相应的虚拟地址翻译过程如下图所示。假设页面命中（在物理内存中），解释翻译步骤 1-5，包括每一步传送的数据是什么、如何得来等。

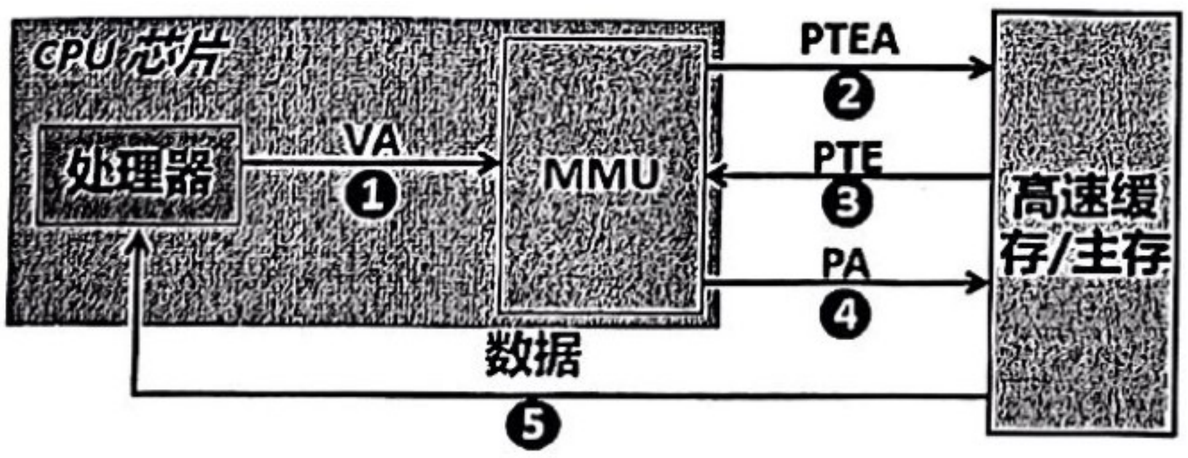

图 3 虚拟地址翻译过程示意图

36. 简述在 shell 命令行中输入“./hello”后回车，系统如何实现 hello 程序从启动运行直至结束的整个过程。