一、单项选择题（每小题 2 分，共 30 分）

1-5 CBAAC

6-10 CACCC

11-15 BDCCC

二、填空题（每空 2 分，共 10 分）

16、 0000FFFAH

17、 00000076H

18、 %rax

19、 0x00b

20、 13.35

三、判断对错（每小题 2 分，共 10 分）

21、× 22、× 23、× 24、× 25、×

四、系统分析题

26、(共 10 分)

答案:（写出任意四个得 8 分，每点 2 分）

1. 被调用者保存，将栈底地址%rbp 的值存在在栈上
2. 比较两个参数的大小
3. 如果小于则跳转
4. 无条件跳转
5. 恢复上一个栈帧，将 rbp 的值出栈

函数：函数会返回两个参数中最小的一个。(2 分)

27、(共 12 分)

答案:

(1) 组索引位 8 位，块内偏移 5 位。

(2) 所以组索引 5 位，组索引对应的位为 0x13。

(3) CT 位为 0x8293 CO 位为 0x0B。

27-详细解答:

（1）64 KB/(8* 32B) = 256 组 Tag 位 37 位，组索引位 8 位，块内偏移 5 位。

（2）VPN 20 位 VPO 12 位。TLBI 128 条目，4 路，一共 32 组。所以组索引 5 位。0x829358B = 1000 0010 1001 0011 0101 1000 1011。所以组索引对应的位为 0x13。

（3）64 KB/(16*32B) = 128 组 组索引位 7 位。块内偏移 5 位。Tag 位 36 位。

所以对应的 0x829358B = 1000 0010 1001 0011 0101 1000 1011

CT 位为 0x8293 CO 位为 0x0B。

28、(共 8 分)

答案:

1）全局符号：x、y、z、main、addvec、addcnt（2 分）

强符号：x、y、main、addvec、addcnt（1 分）

弱符号：z（1 分）

x、y 在.data 节，z 在.bss 节，main.text 节。（1 分）

addvec 在未定义节（UND），addcnt 被优化掉（addvec、addcnt 未说明不扣分）

2）满分 3 分（每点一分，写出任意三点即可）

① 3c 10 60 00　　② 28 10 60 00

③ 30 10 60 00　　④ b3 fe ff ff

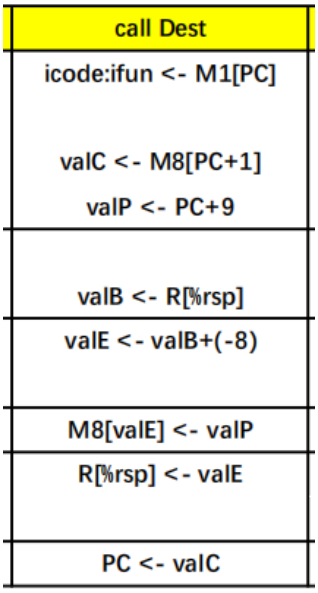
⑤ 2d 09 20 00　　⑥ 23 09 20 00

⑦ a3 fe ff ff

五、综合设计题

29、（共 10 分）

答案：（每空一分）



冒险的种类和应对方法：（冒险种类各一分，解决方法各一分）

1)数据冒险：指令使用寄存器 R 为目的，瞬时之后使用 R 寄存器为源。

处理方法有：

①暂停：通过在执行阶段插入气泡（bubble/nop），使得当前指令执行暂停在译码阶段；

②数据转发：增加 valM/valE 的旁路路径，直接送到译码阶段；

2)加载使用冒险：指令暂停在取指和译码阶段,在执行阶段插入气泡（bubble/nop）

3)控制冒险：分支预测错误：在条件为真的地址 target 处的两条指令分别插入 1 个 bubble。

ret：在 ret 后插入 3 个 bubble。

30、（共 10 分）

答案：（至少 2 种，每种 2 分，最多 4 分）

1）消除不必要的内存引用，减少访存次数。

2）用简单操作替代复杂操作，如移位、加替代乘法/除法。

3）共享公共子表达式，重用表达式的一部分。

4）减少过程调用，消除不必要的跳转指令。

5）循环展开，充分利用计算机的功能单元，充分利用流水线。

程序优化：（6 分）

```
int i, j, k, r;
for(k = 0; k < n; k++)
```

```
{
    for(i = 0; i < n; i++)
    {
        r = A[i][k];
        for(j = 0; j < n - 1; j += 2){
            C[i][j] += r * B[k][j];
            C[i][j + 1] += r * B[k][j + 1];
        }
        for(; j < n; j++)
            C[i][j] += r * B[k][j];
    }
}
```

1）循环展开，充分利用运算单元和流水线。

2）改变循环次序，增加 cache 的命中率。