## 计算机系统概论(2022 秋) 作业 4

1. 某代码由 foo. c 与 bar. c 构成。内容如下:

```
foo. c:
    void p2(void);

int main() {
        p2();
        return 0;
}

bar. c:
    #include <stdio.h>
    extern char main;

void p2() {
        printf("0x%x\n", (int)main);
}
```

- 1) 请问能否正常联合编译?如果可以联合编译则运行结果如何?请说明你给出的回答的理由。
- 2) 如果去掉 bar.c 中的 extern 关键字,能否正常联合编译?如果可以联合编译则运行结果如何?请说明你给出的回答的理由。
- 2. 对于如下代码 foo. c,如果编译为 foo. o,那么在 elf 文件的以下段中,会出现哪些符号? (如果有多种合法分布情况,则任意给出一种即可。)

```
3. extern int add(int a, int b);
4.
5. inline int process(int *v1, int *v2, int *v3) {
6.     *v3 = add(*v1, *v2);
7. }
8.
9. int a[100];
10. extern int b[100];
11. static int c[100];
12.
13. int total process() {
```

```
14. for (int i = 0; i < 100; i++) {
15. process(a+i, b+i, c+i);
16. }
17. }

.bss:
.data:
,text:
.symtab:
```

其中对于 total\_process 函数中用到的符号, 哪些会在编译期被定位, 哪些会在链接期 被定位?

编译期定位的符号:	
链接期定位的符号:	

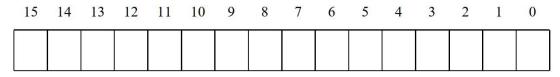
- 3. 下面的问题涉及虚拟地址转换为物理地址:
- 内存是字节可寻址的
- 内存访问是针对 **1 字节的字**(即本题的 word 就是 1 个 byte)
- 虚拟地址 16 位宽 / 物理地址 14 位
- 页面大小为 1024 字节
- TLB 是 4 路组相联 (4-way set associative), 共有 16 个表项。

8	TLB									
Index	Tag	PPN	Valid							
0	8	7	1							
	F	6	1							
	0	3	0							
	1	F	1							
1	1	Е	0							
	2 7	7	0							
	7	3	0							
	$\mathbf{B}$	1	1							
2	0	0	0							
	C	1	0							
	F	8	1							
	7	6	1							
3	8	4	0							
	3	5	0							
	0	D	1							
	2	9	0							

Page Table										
VPN	PPN	Valid	VPN	PPN	Valid					
00	2	0	10	1	1					
01	5	1	11	3	0					
02	7	1	12	9	0					
03	D	1	13	7	1					
04	F	1	14	D	0					
05	3	1	15	5	0					
06	В	0	16	E	1					
07	9	0	17	6	0					
08	7	1	18	1	0					
09	$\mathbf{C}$	0	19	0	1					
0A	3	0	1A	8	1					
0B	1	1	1B	$\mathbf{C}$	0					
0C	0	1	1C	0	0					
0D	D	0	1D	2	1					
0E	0	0	1E	6	1					
0F	1	0	1F	3	0					

在下表中, 所有数字都是十六进制的。前 32 页的 TLB 和页表内容如下:

(1) 下面的框显示了虚拟地址的格式。指出(通过在图上标注)字段(如果存在),这些字段将用于确定以下内容:(如果一个字段不存在,就不要在上面绘制): VPO / VPN / TLBI / TLBT



类似的,在下图标注出物理地址的格式: PPO / PPN

13	12	11	10	9	8	7	6	5	4	3	2	1	0

(2) 对于给定的两个虚拟地址(0x2F09、0x0C53),请分别表示出相应的 TLB 表项和物理地址,并指出 TLB 是否命中、是否发生 page fault。如 page fault,请在"PPN"中输入"-",C 部分留白。

A. Virtual address format (one bit per box)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

B. Address translation

Parameter	Value
VPN	0x
TLB Index	0x
TLB Tag	0x
TLB Hit? (Y/N)	
Page Fault? (Y/N)	
PPN	0x

C. Physical address format (one bit per box)

13	12	11	10	9	8	7	6	5	4	3	2	1	0
												30	

4. 程序运行之前需要为程序分配对应的内存空间,并为内存空间赋予一定的初始值和属性,下表为 x86\_32 架构下用户空间的内存映射关系。请填写下表。并说明可执行文件对应的段(.data, .bss, .text)段分别会被以什么属性映射到哪部分内存空间中

内存起始地址	内存类 型	内存初始值	内存属性
0xFFFFFFF	内核地 址	系统决定	不可读,不可写
0xC0000000	栈		
0x40000000	运行时 堆空间		
由可执行文件决定			可读、可写
0x08048000 保留	内存地址。		

.data:

.bss:

.text: