

C语言综合研究与高强度程序设计训练11

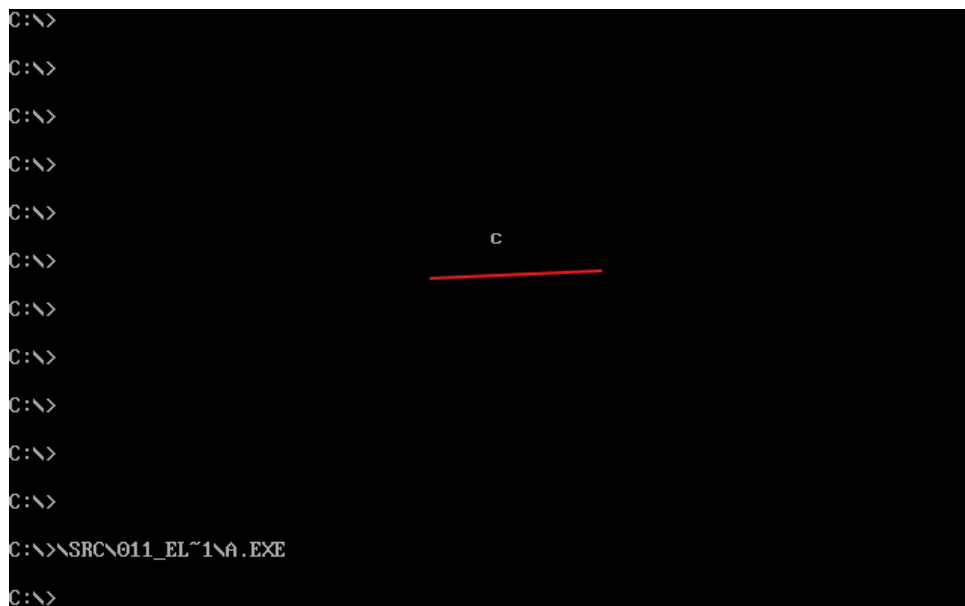
在定义数组a的语句中加入相关的内容使得下面的程序可以在屏幕中间打印一个字符“c”

```
1 char a[200] = {'c'};  
2 main() { ((void(far *)))(long)a(); }
```

- 根据以前学过的操作显存中数据来显示字符可以写出一下程序

```
1 main() { *(char far *)(0xb8000000 + 160 * 10 + 80) = 'c'; }
```

- 运行结果



- 根据书中所给代码可以看到 a 为函数指针，返回值为空，参数为空，所以可以写一个函数功能为在屏幕中打印 c。

```
1 void f();  
2 main() {  
3     f();  
4 }  
5 void f() { *(char far *)(0xb8000000 + 160 * 10 + 80) = 'c'; }
```

- 运行结果



- 然后修改程序查看汇编代码

```

1 char a[200] = "0";
2 void far f();
3 main() {
4     f();
5     ((void(far *)))(long)a();
6 }
7 void far f() { *(char far *) (0xb8000000 + 160 * 10 + 80) = 'c'; }

```

◦ 汇编代码

```

076A:01FA 9A0D026A07 CALL 076A:020D
076A:01FF BB9401 MOV BX,0194
076A:0202 1E PUSH DS
076A:0203 07 POP ES
076A:0204 0E PUSH CS
076A:0205 B90C02 MOV CX,020C
076A:0208 51 PUSH CX
076A:0209 06 PUSH ES
076A:020A 53 PUSH BX
076A:020B CB RETF
076A:020C C3 RET
076A:020D BB00B8 MOV BX,B800
076A:0210 8EC3 MOV ES,BX
076A:0212 BB9006 MOV BX,0690
076A:0215 26 ES:
076A:0216 C60763 MOV BYTE PTR [BX],63
076A:0219 CB RETF

```

- 从中可以看到 `((void(far *)))(long)a()`; 对应的汇编代码转换成了函数从 `1fff->20b` 据此可以推测如果将数组 `a` 中的值改为函数 `f()` 对应的机器码的话就可以正确打印了。

• 修改程序

```

1 char a[200] = "BB00B8EC3BB900626C60763CB";
2 void far f();
3 void(far *b)();
4 main() {
5     f();
6     ((void(far *)))(long)a();
7 }
8 void far f() { *(char far *) (0xb8000000 + 160 * 10 + 80) = 'c'; }

```

- 然后再debug中卡到赋值失败，然后修改赋值方式

```

-d ds:0194
07C1:0190          42 42 30 30 42 38 38 45 43 33 42 42          BB00B8EC3BB
07C1:01A0 39 30 30 36 32 36 43 36 30 37 36 33 43 42 00 00 900626C60763CB..
07C1:01B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:01C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:0200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
07C1:0210 00 00 00 00 .....
-d cs:20d
076A:0200          BB 00 B8          ...
076A:0210 8E C3 BB 90 06 26 C6 07 63 CB C3 55 8B EC EB 0A ...&..c..U...
076A:0220 8B 1E 66 02 D1 E3 FF 97 72 02 A1 66 02 FF 0E 66 ..f.....r..f...f
076A:0230 02 0B C0 75 EB FF 16 5C 02 FF 16 5E 02 FF 16 60 ...u...^....
076A:0240 02 FF 76 04 E8 DA FE 59 5D C3 FB 00 C1 07 00 00 ...v...Yl.....
076A:0250 2E 8F 06 4A 02 2E 8C 1E 4C 02 FC 8E 06 90 00 BE ...J...L.....
076A:0260 80 00 32 E4 26 AC 40 8C C5 87 D6 93 8B 36 8A 00 ..2.&.e.....6..
076A:0270 83 C6 02 B9 01 00 80 3E 92 00 03 72 11 8E 06 8C .....>...r....
076A:0280 00 8B FE B1 7F 32 C0 F2 AE E3 76 80 F1 .....2....v...

```

• 修改程序

```
1 char a[200] = {0xBB, 0x00, 0xB8, 0x8E, 0xC3, 0xBB, 0x90,  
2             0x06, 0x26, 0xC6, 0x07, 0x63, 0xCB};  
3 void far f();  
4 main() {  
5     ((void(far *)))(long)a();  
6 }  
7 void far f() { *(char far *) (0xb8000000 + 160 * 10 + 80) = 'c'; }
```

- 运行结果

- 可以看到数据复制成功

```

-d ds:0194
07C1:0190          BB 00 BB 8E-C3 BB 90 06 26 C6 07 63          .....&..c
07C1:01A0 CB 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:01B0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:01C0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:01D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:01E0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:01F0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:0200 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
07C1:0210 00 00 00 00          .....
- ;

```

- 接程运行一下可以看到打印出c

```
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
c
C:\>
C:\>
C:\>
C:\>
C:\>\SRC\011_EL~1\A.EXE
```