

第十六章习题参考答案

包含题目：习题 16.1、16.2、16.9、16.10 和 16.14

题目 16.1 对于如下程序：

```

1  #include <stdio.h>
2  int main() {
3      int x = 1;
4      int *ptr1;
5      int **ptr2;
6
7      ptr2 = &ptr1;
8      *ptr2 = &x;
9      **ptr2 = 2;
10
11     x++;
12     (*ptr1)++;
13     (**ptr2)++;
14
15     printf ("%d\n", x);
16 }
```

- (1) 程序的输出是什么？提示：ptr2 是一个指向指针的指针。
- (2) 请描述语句 (**ptr2)++; 执行之后的运行时栈中的内容。

解答

- (1) 5[空格][换行]
- (2) 如图所示

0x4000 00BC		
0x4000 00C0	0x4000 00C0(R16)	Ptr1
0x4000 00C4	0x4000 00C4(R17)	Ptr2
0x4000 00C8	0x0000 0005(#5)	x
0x4000 00CC	R30	
0x4000 00D0	R31	<-R30

题目 16.2 下列代码片段的输出是什么？

(1)

```

1  char ch[7] = "1a2b3c";
2  int i, s = 0;
```

```

3  for (i = 0; i < 6; i++) {
4      if (ch[i] >= '0' && ch[i] <= '9')
5          s = s + ch[i] - '0';
6      else
7          s = s + ch[i] - 'a';
8  }
9  printf ("%d\n", s);

```

(2)

```

1  char str[13] = "hello world!";
2  char *p;
3  p = str;
4  while (*p != ' ') {
5      printf ("%c", *p - 'a' + 'A');
6      p++;
7  }

```

解答

(1) 9[换行]

(2) HELLO

题目 16.9 将函数 StringLength 翻译为 DLX 汇编语言。

```

1  int StringLength (char string[]) {
2      int index = 0;
3      while (string[index] != '\0')
4          index = index + 1;
5      return index;
6  }

```

解答

```

1  StringLength:  subi    r29, r29, #4
2                  sw      0(r29), r16 ; 压入r16 (寄存器的保存)
3                  addi    r16, r0, #0 ; r16 <- index
4  LOOP:         addi    r9, r4, r16 ; 计算string[index]的地址值
5                  lb      r10, 0(r9) ; r10 <- string[index]
6                  beqz    r10, EXIT
7                  addi    r16, r16, #1; index = index + 1
8                  J       LOOP
9  EXIT:         addi    r2, r16, #0 ; return index
10                 lw      r16, 0(r29) ; r16出栈
11                 addi    r29, r29, #4
12                 ret

```

R8~R15 和 R24、R25 用于存放临时产生的值，R16~R23 用于存放局部变量。

题目 16.10 如下代码从键盘读入一个字符串，将其中的小写字母转换为大写字母后输出。代码中存在 bug，请找出并修复。

```
1  #include <stdio.h>
2
3  char *ToUpper (char *inchar);
4
5  int main () {
6      char str[10];
7      printf ("Enter a string : ");
8      scanf ("%s", str);
9      printf ("%s \n", ToUpper (str) );
10 }
11
12 char *ToUpper (char *inchar) {
13     char str[10];
14     int i = 0;
15     while (*(inchar + i) != '\0') {
16         if ('a' <= *(inchar + i) && *(inchar + i) <= 'z')
17             *(str + i) = *(inchar + i) - ('a' - 'A');
18         else
19             *(str + i) = *(inchar + i);
20         i++;
21     }
22     return str;
23 }
```

解答 该程序的 bug 为：ToUpper 函数中返回值是一个局部变量的指针，离开该函数后指针所指的内容即为非法；根据题意可知，返回值应该仍然是输入参数。

解决方案：将 `char str[10];` 改为 `char *str = inchar;` 即可。

题目 16.14 如果输入流为 `abc123 def`

对于如下函数调用，字符数组 `x` 的值将是什么？

```
1  scanf ("%s", x)
```

解答 字符数组 `x` 中的值未发生改变，如果 `x` 声明时为初始化，将会是垃圾数据。