

C 语言实现_查看反汇编代码

1	076A:0000 0E 栈	PUSH CS	; 将代码段寄存器 (CS) 的值压入堆栈
2	076A:0001 1F (DS)	POP DS	; 弹出堆栈顶的数据到数据段寄存器 (DS)
3	076A:0002 BA 0E00 为数据段的某个偏移地址	MOV DX,000E	; 将 0x000E 移动到 DX, 设置 DX
4	076A:0005 B4 09 09h (输出字符串)	MOV AH,09	; 设置 DOS 中断 21h 的功能号为
5	076A:0007 CD 21 输出	INT 21	; 调用 DOS 中断 21h, 执行字符串
6	076A:0009 B8 014C 的程序结束功能 (返回码为 1)	MOV AX,4C01	; 将 4C01 移动到 AX, 调用 DOS
7	076A:000C CD 21	INT 21	; 调用 DOS 中断 21h, 结束程序
8	076A:000E 54	PUSH SP	; 将堆栈指针 (SP) 的值压入堆栈
9	076A:000F 68 69 字符 'hi'	DB 68, 69	; 定义两个字节的常量数据, ASCII
10	076A:0011 7320 跳转到 0033	JNB 0033	; 无符号比较下跳转, 如果没有进位则
11	076A:0013 7072 00787	JO 00787	; 如果溢出标志设置, 则跳转到
12	076A:0015 6F	DB 6F	; 定义字节 'o'
13	076A:0016 7261	JB 007A7	; 如果有进位, 则跳转到 007A7
14	076A:0018 6D	DB 6D	; 定义字节 'm'
15	076A:0019 2063 61 AND 操作	AND [BP+DI+611],AH	; 对内存地址中的值与寄存器 AH 做
16	076A:001C 6E	DB 6E	; 定义字节 'n'
17	076A:001D 6E	DB 6E	; 定义字节 'n'
18	076A:001F 6F	DB 6F	; 定义字节 'o'
19	076A:0020 7420	JZ 0042	; 如果零标志设置, 则跳转到 0042
20	076A:0022 62	DB 62	; 定义字节 'b'
21	076A:0023 65	DB 65	; 定义字节 'e'
22	076A:0024 607275 操作	AND [BP+SI+75],DH	; 对内存中的值与寄存器 DH 做 AND
23	076A:0027 20696E 操作	AND [BX+DI+6E],CH	; 对内存中的值与寄存器 CH 做 AND
24	076A:002A 2444 4F	AND [SI+4F],AL	; 对内存中的值与 AL 做 AND 操作
25	076A:002D 206D6F AND 操作	AND [BP+6F],CH	; 对内存地址中的值与寄存器 CH 做
26	076A:0030 53	PUSH BX	; 将寄存器 BX 压入堆栈
27	076A:0031 206D6F AND 操作	AND [BP+6F],CH	; 对内存地址中的值与寄存器 CH 做
28	076A:0040 50	PUSH AX	; 将寄存器 AX 压入堆栈