

## 《第5章函数》习题

### 一、选择题

1. 设有下列变量说明与函数说明，则合法的函数调用语句是\_\_\_\_\_。  

```
int func( int a[], int n );  
int a[10], n, x;
```

A、`x = func( a, n );`  
B、`func( a, n ) = x;`  
C、`x = func( a[], n );`  
D、`x = func( a[10], n );`
2. 以下选项中，合法的函数说明语句是\_\_\_\_\_。  
A、`void func( char *a, char b[] );`  
B、`void func( char a[], b[] );`  
C、`void func( char a[]; char *b );`  
D、`void func( char *a, *b );`
3. 调用函数时，基本类型变量作函数实参，它和对应的形参\_\_\_\_\_。  
A、各自占用独立的存储单元  
B、共占用一个存储单元  
C、同名时才能共用存储单元  
D、不占用存储单元
4. 有以下程序片断  

```
float fun(int x, int y)  
{  
    return (x+y);  
}  
main()  
{  
    int a=2,b=5,c=8;  
    printf("%3.0f\n", fun((int)fun(a+c,b), a-c));  
}
```

程序运行\_\_\_\_\_。

A、编译出错  
B、输出 9  
C、输出 21  
D、输出 9.0
5. 一个C语言程序中可以包含多个函数，以下选项中，正确的是\_\_\_\_\_。  
A、函数的定义可以嵌套，但函数的调用不可以嵌套  
B、函数的定义不可以嵌套，但函数的调用可以嵌套  
C、函数的定义和函数的调用均可以嵌套  
D、函数的定义和函数的调用均不可以嵌套
6. 若有函数调用语句：  

```
func(rec1, rec2+rec3, (rec4, rec5));
```

则可以判断该函数的参数个数是\_\_\_\_\_。

- A、3  
B、2  
C、5  
D、有语法错误
7. 已知函数原型 `int f(int)`和 `int g(int)`。下列语句中，正确调用函数的语句是\_\_\_\_\_。
- A、`int f(3);`  
B、`f( int g(3) );`  
C、`int g( f(3) );`  
D、`p=f( g(3) +1 );`
8. C 程序中，`return` 语句的作用是\_\_\_\_\_。
- A、终止程序运行  
B、返回到上层循环  
C、返回到外层结构  
D、返回到上层函数
9. 在一个被调用函数中,关于 `return` 语句使用的描述，错误的是\_\_\_\_\_。
- A、被调用函数中可以不用 `return` 语句  
B、被调用函数中可以使用多个 `return` 语句  
C、被调用函数中,如果有返回值,就一定要有 `return` 语句  
D、被调用函数中,一个 `return` 语句可返回多个值给调用函数
10. 以下叙述中不正确的是\_\_\_\_\_。
- A、在一个函数内的复合语句中定义的变量在本函数范围内有效  
B、在不同的函数中可以使用相同名字的变量  
C、函数中的形式参数是局部变量  
D、在一个函数内定义的变量只在本函数范围内有效
11. 以下不正确的说法是\_\_\_\_\_。
- A、实参可以是常量、变量或表达式  
B、实参可以是任何类型  
C、形参可以是常量、变量或表达式  
D、形参应与对应的实参类型一致
12. 以下正确的说法是\_\_\_\_\_。
- A、实参与其对应的形参共同占用一个存储单元  
B、实参与其对应的形参各占用独立的存储单元  
C、只有当实参与其对应的形参同名时才占用一个共同的存储单元  
D、形参是虚拟的，不占用内存单元
13. 在 C 语言程序中，若对函数类型未加显式说明，则函数的隐含类型为\_\_\_\_\_。
- A、`void`  
B、`double`  
C、`char`  
D、`int`
14. 要在 C 语言程序中使用系统函数 `sqrt()`，需要使用 `include` 命令包含\_\_\_\_\_库文件
- A、`#include<math.h>`  
B、`#include<string.h>`  
C、`#include<io.h>`  
D、`#include<stdio.h>`

15. 当调用函数时，实参是一个数组名，则向函数传递的是\_\_\_\_\_。

- A、数组的长度
- B、数组的首地址
- C、数组每一个元素的地址
- D、数组每一个元素中的值

16. 下列程序的运行结果是\_\_\_\_\_。

```
#include <stdio.h>
```

```
int b=1;
```

```
int fun(int x)
```

```
{
```

```
    static int b=3;
```

```
    b+=x;
```

```
    printf("%d ",b);
```

```
    return b;
```

```
}
```

```
int main()
```

```
{
```

```
    int a=2;
```

```
    printf("%d\n",fun(a+fun(b)));
```

```
    return 0;
```

```
}
```

A、4 10 10

B、6 6 8

C、3 3 9

D、10 10 4

## 二、程序填空题

1. 以下函数 `int sum(int x[], int num)` 计算数组 `x` 前 `num` 个元素之和。在主函数中，输入 10 个任意整数和下标 `index1`、`index2` 的值（设  $1 \leq \text{index1} \leq \text{index2} \leq 9$ ），调用 `sum()` 函数计算 `index1` 元素到第 `index2` 元素的和，并输出结果。

如输入：

```
1 3 2 5 7 9 6 4 8 10
```

```
2 4
```

则输出：Sum=14

例示说明：下标从 2 至 4 的元素为 2、5、7，其累加为 14

源程序：

```
#include <stdio.h>
```

```
int sum( int x[], int num )
```

```
{
```

```
    int i, s=0;
```

```
    for ( i=0; ____ (1) ____; i++ )
```

```
        s = ____ (2) ____;
```

```
    return s;
```

```

}
void main()
{
    int i, index1, index2, result;
    int x[10];
    for ( i=0; i<10; i++ )
        scanf( "%d", &x[i] );
    scanf( "%d%d", &index1, &index2 );
    result = sum( x+index1, ____ (3) ____ );
    printf( "Sum=%d\n", result );
}

```

2. 输入一个整数，逆序后输出。函数 reverse(int number)的功能是返回 number 的逆序数。例如 reverse(12345)的返回值是 54321。

运行示例：

输入整数：-1230

逆序后：-321

源程序：

```

#include <stdio.h>
int reverse(int);
void main( )
{
    int in;
    ____1____;
    printf("输入整数:");
    scanf("%d", &in);
    m = ____2____;
    printf("逆序后:%d\n",m);
    getchar(); /*暂停程序，按任意键继续*/
}
int reverse(int number)
{
    ____3____;
    int res=0;
    flag = number < 0 ? -1 : 1; /* flag 标志正负数 */
    if(number < 0) number = - number;
    while(number != 0)
    {
        res = res*10 + number%10;
        ____4____;
    }
    return ____5____*res;
}

```

3. 输入一个长度小于 40 的字符串到数组 s，并复制到数组 t 中，再将 s 中的字符串逆序连接到 t 的后面并输出。例如：输入"ABCD"，则字符串 t 为"ABCDDCBA"。

源程序：

```

#include <stdio.h>

```

```

#include <string.h>
void main()
{
    char s[80],t[80];
    int i;
    ____1____;
    printf("输入字符串: ");
    scanf("%s",s);
    m=strlen(s);
    for(i=0; ____2____;i++)
        t[i]=s[i];
    for(i=0;i<m;i++)
        t[m+i]=s[____3____];
    t[m+i]='\0';
    printf("新的字符串: ____4____\n",t);
    getchar(); /*暂停程序, 按任意键继续*/
}

```

### 三、编程题

1. 输入 2 个正整数  $m$  和  $n$  ( $m \geq 1, n \leq 10\,000$ ), 输出  $m \sim n$  之间所有的 Fibonacci 数。Fibonacci 数列(第一项起): 1,1,2,3,5,8,13,21,...。要求定义并调用函数 `fib(n)`, 它的功能是返回第  $n$  项 Fibonacci 数。例如, `fib(7)` 的返回值是 13。
2. 输入 2 个正整数  $m$  和  $n$  ( $1 \leq m, n \leq 1000$ ), 输出  $m \sim n$  之间的所有水仙花数。水仙花数是指各位数字的立方和等于其自身的数。要求定义并调用函数 `is(number)` 判断 `number` 的各位数字之立方和是否等于其自身。
3. 输入两个由整型数构成的集合(元素个数均为 5) 分别放到数组 A 和 B 中, 求这两个集合的交集(交集的元素由两个集合中的相同元素构成), 最后输出交集的元素, 要求在主函数中输入 A 和 B 数组, 并输出交集的元素, 在被调函数中实现求这两个集合的交集。
4. 输入 10 个整数(存入数组 a), 再输入整数 x; 要求编写函数实现将该数组元素向左偏移 x 个位置后循环输出。要求在主函数中输入 a 数组, 并输出最后后的结果, 在被调函数中实现循环左移 x 个位置。
5. 将两个已按升序排列的数组合并成一个升数组, 要求在主函数中输入两个数组, 并输出合并后的结果, 在被调函数中实现合并。
6. 编写程序, 将字符串中的第 m 个字符开始的全部字符复制到另一个字符串中。要求在主函数中输入字符串及 m 的值, 并输出复制结果, 在被调函数中完成复制。
7. 输入任意一行字符串(设字符数不大于 80), 提取该字符串中的数字字符并组成一个整数, 输出该整数及其两倍的值。要求在主函数中输入字符串, 并输出复制结果, 在被调函数中完成提取该字符串中的数字字符并组成一个整数。
8. 从键盘输入一行 16 进制字符串(无空格, 字母必须大写), 将其转换为 10 进制(遇到非 16 进制字符结束), 并输出转换结果。要求在主函数中输入 16 进制字符串, 并输出结果, 在被调函数中完成转换。
9. 输入一个长度小于 80 的字符串, 将小写字母转换为大写, 如果输入串没有小写字母, 则输出提示信息。要求在主函数中输入字符串, 并输出结果, 在被调函数中完成将小写字母转换为大写。

10. 输入一个整数，将其逆序输出。要求在主函数中输入整数，编写一个递归函数实现将该整数逆序输出。例如在主函数中输入 123456，则在输出 654321。
11. 编写递归程序，实现字符串的倒序输出。例如输入 “123xyz”， 则输出 “zyx321”。