

2021《计算导论与程序设计》期中考试参考答案

一、选择题 (每题 2 分, 共 20 分)

1、下面针对 C 语言中函数的描述, 正确的有(B)句

- (1) 调用函数时, 只能将实参的值传递给形参, 形参的值不能传递给实参。
- ~~(2) 主函数必须在其他函数之前定义, 函数内不可以再定义函数。~~
- ~~(3) 函数必须有返回值。~~
- (4) 不同的函数中可以使用相同的变量名。

A. 1 B. 2 C. 3 D. 4

2、十进制数 123 转换为七进制数是(B)

A. 233 B. 234 C. 235 D. 236

3、设函数 fun 的原型为 void fun(char, int, double), 则以下对函数 fun 的调用语句中, 正确的是(A)

- A. fun(10, 'A', 23);
- ~~B. fun("def", 65, 3.14);~~
- ~~C. float a; a=fun('C', 99, 6.5);~~
- ~~D. fun(F, 7, 2.8);~~

4 定义 char c='A', short s=3, 则表达式 c+s/2 的计算结果是什么类型的数据 (C)

A. char B. short C. int D. double

5、NS 图 and 传统程序流程图相比, 优点是(B)。

- A. 执行过程可随意跳转
- B. 保证算法的单入单出特性。
- C. 篇幅较长
- D. 选择结构和循环结构符号更直观

6、假设从左到右对输入数据 1、2、3、4 进行多次入栈和出栈操作, 最后不可能得到的输出组合是 (D)

A. 4321 B. 1234 C. 2431 ~~D. 3412~~

7、下面关于递归的说法中错误的是 (C)

- A. 递归算法通常是一个多分支结构
- B. 递归执行过程体现了“自顶向下, 逐步细化”的思想
- C. 递归的书写通常比迭代简洁, 执行效率也比迭代高
- D. 能用递归方法解决的问题也可以用迭代方法解决

8、假设表达式 exp 定义如下: (1) 常量、变量是表达式; (2) 如果 exp 是表达式, 则(exp)也是表达式; (3) 若 Θ 是运算符, exp1 和 exp2 是表达式, 则 $\text{exp1exp2}\Theta$ 也是表达式; (3) 有限重复上述步骤, 仍然是表达式。假设 a、b、c 是变量, Θ 定义为 +、-、*、/, 请问下列表达式正确的有(B)个

- (1) $5*b+c$
 - (2) $a*(2c+)$
 - (3) $b3*a+$
 - (4) $6(c8-)/$
- A. 1 B. 2 C. 3 D. 4

9、假设 $\text{int } a=3, b=4$ ，请写出表达式 $(a>b)+5-(a=b)$ 的值(A)

A. 1 B. 5 C. 2 D. 6

10、以下程序的输出结果是(B)

```
long fun( int n)
{
    long s;
    if(n==1 || n==2)
        s=2;
    else
        s=n-fun(n-1);
    return s;
}
int main()
{
    printf("%ld\n", fun(5));
    return 0;
}
```

A. 1 B. 2 C. 3 D. 4

二、填空题 (每空 2 分，共 20 分)

说明：填空题前三题，每个空只要意思相近、逻辑合理均可给分。

1、程序设计语言通常包含四种成分，分别是数据成分/数据、运算成分/运算、控制成分/控制和传输成分。

2、C 语言中函数参数的传递方式只有值传递/按值传递，但可以通过其它方式实现引用/按引用/地址/按地址传递的效果。

3、函数 A 调用函数 B，当函数 B 运行结束后，之所以能正确返回到函数 A，是因为在函数 B/b 的 活动记录/运行环境 里保存了函数 A 的返回地址。

4、已知一个四位正整数 ABCD，满足 $ABCD*4=DCBA$ ，求出满足条件的这个四位数。请补充下面 main 函数中缺失的语句，并使得循环次数尽可能少。

```
int main()
{
    int num, copyNum, reverseNum;
    int notFound;

    notFound = 1;
    for(num=1001; num<=2499 (或 num<=9999) && notFound; num++) // 注：没有
notFound 扣 1 分
```

```

{
    copyNum = num;
    reverseNum = 0;
    while(____copyNum (或 copyNum!=0)____) //求 num 的逆数，保存在 reverseNum
中。
    {
        reverseNum = ____reverseNum*10+copyNum%10____;
        copyNum = ____copyNum/10____;
    }
    if (____num*4==reverseNum____)
        notFound= 0;
}
if (notFound)
    printf("不存在这样的四位数\n");
else
    printf("该四位数为: %d\n", num-1);

return 0;
}

```

5、假设有如下程序：

```

int fun (int a, int b)
{
    a=a+3;
    b=a/b;
    printf("a=%d b=%d\n", a, b);
    return b;
}

```

```

int main( )
{
    int a=1, b=3;
    b=fun(a, b);
    switch (b)
    {
        case 1: printf("E");
        case 2: printf("D");
        case 3: printf("G");
        default: printf("NB\n");
    }
    printf("a=%d b=%d\n", a, b);
}

```

请给出程序执行的结果：

1. a= 4 b= 1

2. EDGNB
3. a=1 b=1

三、简答题 (共 40 分)

说明：简答题只需要检查答案中是否有给分点，只要有给分点均可给分。如果没有，陈述再多也不给分。

1. 数据类型包含哪三个要素？为什么 C 语言在定义变量的时候需要给出它的数据类型？(8 分)

参考答案：

数据类型三要素：逻辑结构(数据的抽象)(1 分)、存储结构(数据的存储)(2 分)、数据的基本操作(1 分)。

C 语言变量定义需要给出数据类型原因：(1) 需要根据数据类型确定变量的存储空间(3 分)；(2) 需要根据数据类型判断对变量的运算操作是否合法(1 分)。

2. 以下是计算 $1 + 2 + 3 + \dots + 100$ 的程序，请用数学归纳法证明该程序的正确性。(8 分)

```
#include <stdio.h>
#define N 100
int main()
{
    int i;
    int sum; //存放求得的和

    i = 2;
    sum = 1; //初始化
    while(i <= N){
        sum = sum + i;
        i = i + 1;
    }

    printf("sum is:%d\n",sum);
    return 0;
}
```

参考答案：

设 $a(n) = 1 + 2 + 3 + \dots + n$ ，对于以上循环程序，我们证明变量 sum 在每次循环前始终有 $sum(n) = a(n)$ ($n \geq 1$) 成立。

(1) 当 $n=1$ 即第一次循环前，此时 sum 初始化为 1，即 $sum(1) = a(1) = 1$ 显然成立。

(2) 假设第 k 次循环前 $sum(k) = a(k)$ 成立 ($k \geq 1$)。由于变量 i 初始值为 2，每次循环后 i 的值加一，因此第 k 次循环前 $i = k + 1$ 。在本次循环中执行语句 $sum = sum + i$ ，相当于执行 $sum(k+1) = sum(k) + k + 1 = a(k) + k + 1 = a(k+1)$ ，所以本次循环执行完后， $sum(k+1) = a(k+1)$ 仍然成立。

(3) 当 $k=100$ ，即 $i=101$ 时循环终止，此时 $sum = sum(100) = a(100) = 1 + 2 + 3 + \dots + 100$ ，因此程

序结果执行正确。

3. 对于一个长度不超过 9 位的自然数 num ，假定从个位起代表第一个奇数位，请设计一个函数 $check$ ，用于判定 num 中各个数字的奇偶性是否和它所在位置的奇偶性全部保持一致。比如 521 中数字 5(奇)、2(偶)、1(奇)与所在位置的奇偶性全部保持一致(括号中代表位置奇偶性)，837 中数字 8(奇)、3(偶)、1(奇)没有全部保持一致。对于满足要求的 num ，返回值 1，否则返回 0。(共 24 分)

问题：

(1) 请用 C 语言设计 $check$ 函数的原型，注意参数的数据类型尽量合理。(6 分)

参考答案：

$int\ check(int\ num, int\ isodd);$ // num 和 $isodd$ 等参数名字可以不写。

评分要点：函数名字 $check$ 1 分；参数 num 代表待检测的自然数，可以为 $int/long$ 等类型，2 分；参数 $isodd$ 代表当前 num 的个位所处位置是否为奇数位，可以为 $char/short/int$ 等类型，2 分；返回值可以为 $char/short/int$ 等类型，1 分。注：该题的函数原型中参数个数和类型可以有多个答案，但必须和第(2)问的递归定义表达式中的参数保持一致，否则扣一半的分。

(2) 给出 $check$ 函数的递归定义表达式(可用数学公式或伪代码)。(8 分)

参考答案：

$check(num, isodd)$

$$= \begin{cases} num \text{ MOD } 2 == isodd \text{ MOD } 2 & \text{if } num/10 = 0 \\ check(num/10, not\ isodd) \text{ AND } (num \text{ MOD } 10 \text{ MOD } 2 == isodd \text{ MOD } 2) & \text{if } num/10 > 0 \end{cases}$$

评分要点：第一个分支：临界条件 3 分，其中操作 2 分，if 条件 1 分；第二个分支：递推表达式 5 分，其中递归调用 $check(num/10, not\ isodd)$ 2 分；判断个位数值与位置的奇偶性是否一致，即 $num \text{ MOD } 10 \text{ MOD } 2 == isodd \text{ MOD } 2$ ，2 分；if 条件 1 分。

评分要点：

- (1) 算法可以用数学公式、伪代码、C 语言或自然语言表示，只要意思一致即可
- (2) $check$ 函数必须为递归表达，如果写成了非递归表达，最高分为 4 分。
- (3) 如果学生有新的递归算法思路，只要逻辑合理，均可给分。

(3) 利用伪代码设计 $check$ 函数的“非递归”算法。(10 分)

注：伪代码可采用如下表示

循环结构 $while() \text{ do} \dots$ ，分支结构 $if() \text{ then } \dots \text{ else} \dots$ ，复合操作 $\{ \dots \}$

赋值操作可用 \rightarrow 或 \leftarrow

逻辑运算中与、或、非可用 and, or, not

变量声明可用 $char, int, real$ 等数据类型

参考答案：

```
int check(num, isodd) // isodd 标记当前 num 的个位所处位置是否为奇数位
{
    int flag=1; // 变量定义 1 分
    while (num && flag) do // 循环条件 2 分，每个条件项 1 分
    {
        if (num MOD 10 MOD 2 != isodd MOD 2) // 判断条件 2 分，
            then flag=0; // 修改标记变量 1 分

        num ← num /10; // 改变 num 值，1 分
    }
}
```

```
        isodd ← not isodd; // 将 isodd 值取反, 2 分
    }
    return flag; // 返回值 1 分
}
```

评分要点:

- (1) 算法可以用伪代码、NS 图、C 语言或自然语言表示, 只要意思一致即可
- (2) check 函数必须为“非递归”算法, 如果写成了递归算法, 最高分为 5 分。
- (3) 按照以上评分点给分, 如果学生描述的算法的某个步骤包含多个给分点, 给出对应的分数总和。
- (4) 如果学生有新的算法思路, 只要逻辑合理, 均可给分。