



模 块 化



【实习目标】

- 能够按照模块化的思想对问题进行分解,简化问题求解过程。
- 能够针对模块化分解后的问题设计出与程序设计语言无关的求解算法。
- 能够使用 C++ 语言编写模块化程序解决实际问题。



5.1 课 程 实 习

1. 编写程序: 计算 $1/1^m + 1/2^m + \dots + 1/n^m$, 其中 m 为大于 0 的整数。

(1) 对问题进行分解。



(2) 设计求解该问题的算法。

步骤	处 理

(3) 用 C++ 语言写出实现该算法的程序核心代码。

--

- (4) 上机调试并测试你的程序。
2. 编写程序验证组合公式： $C(2n, n)=C(n, 0)^2+C(n, 1)^2+C(n, 2)^2+\cdots+C(n, n)^2$ 。
- (1) 将问题进行分解。

- (2) 设计求解该问题的算法。

步骤	处 理



(3) 用 C++ 语言写出实现该算法程序的核心代码。

(4) 上机调试并测试你的程序。

3. 已知数列 $a_i = 2a_{i-1} + 3a_{i-2}$, 且 $a_1 = a_2 = 1$, 求第 n 项的值(要求: 设计递归算法求解)。

(1) 将问题进行分解。

(2) 设计求解该问题的算法。

步骤	处 理

(3) 用 C++ 语言写出实现该算法程序的核心代码。



(4) 上机调试并测试你的程序。

4. 如果一头母牛从出生起第 4 个年头(即出生后 3 年)开始每年生一头母牛,按此规律,第 1 年只有 1 头刚出生的母牛(即该母牛第 4 年开始每年生一头母牛),到第 n 年时有多少新出生的母牛?(要求:设计递归算法求解)

(1) 将问题进行分解。

--

(2) 设计求解该问题的算法。

步骤	处 理

(3) 用 C++ 语言写出实现该算法程序的核心代码。

(4) 上机调试并测试你的程序。

5.2 课 后 习 题

一、算法设计

1. N 名学生排成两行, M 名女生排在第一行、 $N-M$ 名男生排在第二行, 问共有多少种排法?

2. 求满足如下条件的 m : m 、 m^2 和 m^3 均为回文数且 $10 < m < 1000$ 。注: 一个数是回文数, 则该数的值与其倒序数的值相同, 例如 121、12321 等都是回文数。

3. 广义水仙花数问题: 一个 3 位数 abc 如果满足 $abc = a^3 + b^3 + c^3$, 则这个数称为水仙花数。如果一个 N 位数所有数码的 N 次方的和加起来等于这个 N 位数本身, 则这个数称为广义水仙花数。显然, 水仙花数就是 $N=3$ 的广义水仙花数。求所有满足 $N=m$ 的广义水仙花数。

4. 猴子吃桃问题: 一只猴子摘下若干个桃子, 第 1 天吃了一半, 还不够, 又多吃了一个; 第 2 天将剩下的桃子吃了一半, 又多吃了一个。以后每天都这样吃, 到第 10 天就只剩下一个桃子了。问该猴子共摘下多少个桃子?

5. 八皇后问题: 该问题由 19 世纪著名的数学家高斯 1850 年提出。在一个 8×8 格的国际象棋棋盘上放置 8 个皇后, 使其不能互相攻击, 即任意两个皇后都不能处于同一行、同一列或同一斜线上, 如图 5-1 所示是八皇后问题的一种摆法, 阴影格是放置皇后的格子。问共有多少种摆法?

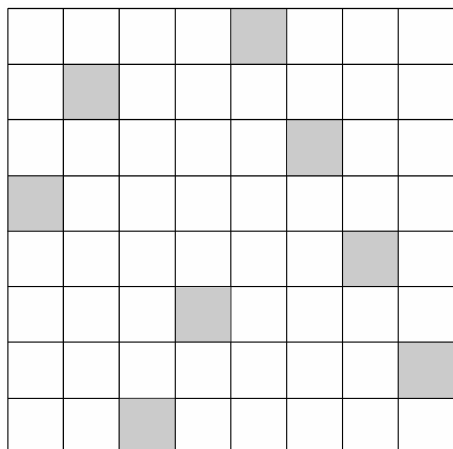


图 5-1 八皇后问题示例

6. 汉诺塔问题：有 3 根相邻的柱子，标号为 A、B、C，A 柱子上从下到上按金字塔状叠放着 n 个不同大小的圆盘，目标是把所有盘子都移动到柱子 B 上，图 5-2 为 $n=3$ 的情况。每次只能移动一个盘子，并且盘子移动过程中同一根柱子上都不能出现“大盘子在小盘子上方”的情况，移动过程中允许将盘子放在柱子 C 上，问至少需要移动多少次？

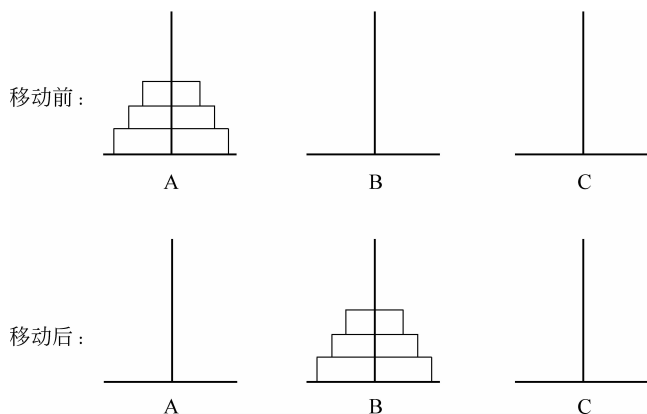


图 5-2 汉诺塔问题示例

二、提高 C++ 语言程序设计能力练习

1. 填空题

(1) 下面是求一个数的绝对值的函数定义，函数名为 `abs`，请将函数定义补充完整。

```

_____
{
    int a;
    a = x > 0 ? x : -x;
}
    
```



```

    return a;
}

```

(2) 下面的程序会在屏幕上输出 5, 请将下面的程序补充完整。

```

#include <iostream>
using namespace std;

_____

int main()
{
    int a=10, b=5;
    cout<<fun(a,b)<<endl;
    return 0;
}

_____

{
    return x<y?x:y;
}

```

(3) 下面程序的功能是计算 $s=1-1/2!+1/3!+\dots+(-1)^{n+1}/n!$, 请将下面的程序补充完整。

```

#include <iostream>
using namespace std;
double fac(int n)
{
    int i, r=1, f=1;
    for (i=1; i<=n; i++)
        _____;
    if (_____)
        f=-1;
    return (double)f/r;
}

int main()
{
    int n, i;
    double facSum=0.0;
    cout<<"请输入 n 的值:";
    cin>>n;
    for(i=1; i<=n; i++)
        _____;
    cout<<facSum<<endl;
    return 0;
}

```

(4) 下面的程序求斐波那契数列前 5 项的和, 请将程序补充完整(斐波那契数列前两项的值为 1, 从第 3 项开始, 每一项的值是前两项的和)。



```
#include <iostream>
using namespace std;
int Fib(int n)
{
    int pre2, pre1=1, cur=1, i=2;
    if (_____)
        return 1;
    do
    {
        i++;
        pre2=pre1;
        pre1=cur;
        _____;
    } while (i!=n);
    return cur;
}
int main()
{
    int i=0, sum=0;
    for (i=1; i<=5; i++)
        sum += _____;
    cout<<sum<<endl;
    return 0;
}
```

(5) 下面程序输出 3~100 的所有素数,请将下面的程序补充完整。

```
#include <iostream>
using namespace std;
int k=0;
bool IsPrime(int n)
{
    int i;
    for(i=2; i<n; i++)
        if(n%i==0)
            _____;
    if(i==n)
        return true;
    return false;
}
int main()
{
    int n;
    cout<<"3~100 之间的素数包括:"<<endl;
    for(n=3;n<100;n+=2)
        if (_____)
```