

编程基础 5.11 习题

1、略

2、九九乘法表

- 方法1

```
1 void showMultiplicationTable1(){
2
3     for (int i = 1; i <= 9; i++) {
4         for (int j = 1; j <= i; j++) {
5             cout << j << "*" << i << "=" << i * j << "\t";//水平制表符
6         }
7         cout << endl;
8     }
9 }
```

```
1 1*1=1
2 1*2=2 2*2=4
3 1*3=3 2*3=6 3*3=9
4 1*4=4 2*4=8 3*4=12 4*4=16
5 1*5=5 2*5=10 3*5=15 4*5=20 5*5=25
6 1*6=6 2*6=12 3*6=18 4*6=24 5*6=30 6*6=36
7 1*7=7 2*7=14 3*7=21 4*7=28 5*7=35 6*7=42 7*7=49
8 1*8=8 2*8=16 3*8=24 4*8=32 5*8=40 6*8=48 7*8=56 8*8=64
9 1*9=9 2*9=18 3*9=27 4*9=36 5*9=45 6*9=54 7*9=63 8*9=72 9*9=81
```

- 方法2

```
1 void showMultiplicationTable2(){
2
3     for (int i = 9; i >= 1; i--) {
4         for (int j = 9; j > 9-i; j--) {
5             cout << j << "*" << i << "=" << i * j << "\t";
6         }
7         cout << endl;
8     }
9 }
```

```
1 9*9=81 8*9=72 7*9=63 6*9=54 5*9=45 4*9=36 3*9=27 2*9=18 1*9=9
2 9*8=72 8*8=64 7*8=56 6*8=48 5*8=40 4*8=32 3*8=24 2*8=16
3 9*7=63 8*7=56 7*7=49 6*7=42 5*7=35 4*7=28 3*7=21
4 9*6=54 8*6=48 7*6=42 6*6=36 5*6=30 4*6=24
5 9*5=45 8*5=40 7*5=35 6*5=30 5*5=25
6 9*4=36 8*4=32 7*4=28 6*4=24
7 9*3=27 8*3=24 7*3=21
8 9*2=18 8*2=16
9 9*1=9
```

- 方法(目前想不出来了)

1 |

1 |

3、略吧()

4、递归+循环 poly()

递归

```
1 double poly(int x ,int n){
2
3     if (n == 0)
4     {
5         return 1;
6     }
7     else if (n==1)
8     {
9         return x;
10    }
11    else if (n>1){
12        return ((2 * n - 1) * x * poly(x, n - 1) - (n - 1) * poly(x, n -
13    2)) / n;
14    }
15 }
```

循环

1 | 想不出来啦555

5、斐波那契数列邻项之比（好像没要求要编程啊哈哈哈哈哈哈）

那就不做啦

6、辗转相减法求最大公约数

递归

```
1  int gcd(int m,int n ) {
2      if (m==n)
3      {
4          return m;
5      }
6      else if (m<n)
7      {
8          return gcd(m, n - m);
9      }
10     else {
11         return gcd(m - n, n);
12     }
13 }
```

循环

```
1 |
```

7、求Sin(x)

```
1  double tsin(double x)
2  {
3      double res = 0; //结果 也就是Sin(x)
4
5      double ele = x; //每一项
6
7      int n = 1; //项数
8
9      do {
10         res += ele;
11         n++; //循环一次，项数 +1
12         ele = -ele * x * x / (2 * n - 1) / (2 * n - 2); //这个的意思是每一项等于
//前一项乘两个x再除以(2 * n - 1) * (2 * n - 2); 有点递推的意味，多想一下，♥
13
14     } while (fabs(ele) >= 1e-6); //每一项绝对值小于10^-6
15
16     return res; //返回结果
17 }
```

8、求素数

(1)略吧哈啊哈哈

略略略

(2)判断素数的算法

```
1  bool isPrime(int num) {
2
3      if(num == 1 || num == 2) return 1; //若数为1||2, 则为质数, 返回TRUE, 即1
4
5      for(int i = 2; i < num / 2; i++) { //这里i<num/2是, 大于num的一半过后 num%i 肯
        定不为0 ,无意义
6          if(num % i == 0 ) return 0;
7      }
8
9      return 1;
10 }
```

(3)求1000以内的质数

```
1  getPrime(int n){
2
3      n==1000;
4
5      for (int i = 1; i < n; i++)
6      {
7          if (i == 1)
8          {
9              continue;
10         }
11         int flag = 0;
12         for (int j = 1; j <= i; j++)
13         {
14             if (i % j == 0)
15             {
16                 flag++;
17             }
18         }
19         if (flag <= 2)
20         {
21             cout << " " << i;
22         }
23     }
24 }
```