

# $F(g)$ 的求解，编程和测试

赵建华

# 原理

- 假设把 $g$ 表示成为互不相交的轮换的乘积  $r_1 * r_2 * \dots * r_k$
- 如果 $x$ 等于 $r_i$ 中的某个元素 $r_{ij}$ , 那么 $g(x)$ 等于 $r_i$ 中的后一个元素 $r_{i(j+1)}$
- 如果某个赋值方式 $V$ 在 $g$ 置换下不变, 当且仅当对于任意的 $x$ ,  $V(x)=V(g(x))$ , 当且仅当, 对于任意的 $r_{ij}$ , 那么 $V(r_{ij})=V(r_{i(j+1)})$ 。
- 因此, 在同一个轮换中的赋值应该相同!

# 原理

- 对于 $n$ 个变量， $m$ 种赋值的情况
  - 总数值就是 $m$ 的 $k$ 次方， $k$ 是轮换的次数
- 对于 $n$ 个变量中取 $m$ 个元素进行赋值的情况
  - 相当于从 $k$ 个轮换中选取一些轮换，使得这些轮换的元素个数总和等于 $m$

# 编程

- 分成2步
  - 求轮换
    - `ROTATE* getR(DISPLACEMENT g);`
    - 参数是置换；返回轮换的信息
    - 要求
      - 不改变任何非局部变量的值
      - 返回的ROTATE指针指向新申请的空间；
  - 求 $F(g)$ : `int getFg(int m, ROTATE *)`;
    - 参数是轮换，以及值m
    - 返回 $F(g)$
    - 要求
      - 不改变任何非局部变量的值

# 测试要求

- 使用gtest分开测试两个函数
- 对于第一个函数
  - 要求测试轮换表示和原来的置换表示的一致性
  - 因此需要编写检测一致性的函数
- 对于第二个函数
  - 要求设计10个以上的轮换以及正确的答案

如何考虑N中去m的情况？