代码（Golang）：

```
package main
import (
    "crypto/rand"
    "fmt"
    "math"
    "math/big"
    "strconv"
)
func main() {
    var count int64 //统计冲突次数
    var sample = 1000 //试验次数
    var rate float64
    var sampleRange int64
    sampleRange = int64(math.Pow(2,32)) //2^32 的范围内取取随机数
    count = match(sampleRange,sample) //计算实验中的重复次数
    fmt.Printf("count = %d\n",count)
    rate = (float64(count)/float64(sample))*100
    fmt.Printf("rate is %.3f\n",rate)
}
func match(Range int64 ,sample int) int64 { //把 t 和试验次数传进参数
    var count int64 = 0 //计重复次数
    var nums [80000]int
    for i := 0; i < sample; i++ {//重复统计
        nums = Rond(Range)//生成随机数组
        //开始判断是否有相同的数，相同则记一次
        if containsDuplicate(nums) == true{
            count += 1
        }
    }
    return count
}
func Rond(Range int64) [80000]int{//t= 80000
    var arr[80000] int
    for i := 0; i < 80000; i++{
        result, _ := rand.Int(rand.Reader, big.NewInt(Range))
        number := result.String()
        //fmt.Println(number)//打印随机生成的
        num, err := strconv.Atoi(number)
        if err == nil{
            //fmt.Printf(" ")
        }
        arr[i] = num
```

```
    }
    return arr
}
func containsDuplicate(num [80000]int) bool {//判断是否冲突
    set := map[int]struct{}{}
    for _ , v := range num {
        if _, has := set[v]; has {
            return true
        }
        set[v] = struct{}{}
    }
    return false
}
```

代码上传地址：

https://gitee.com/shan-yitian/golang/blob/master/Birthday%20Paradox.go

实验结果：

	A	B	C	D	E
1	t (0-2^32)	Percent of conflicts1	Percent of conflicts2	Average	
2	10000	1.10%	1.20%	1.15%	
3	20000	4.10%	4.70%	4.40%	
4	30000	9.50%	8.30%	8.90%	
5	40000	17.80%	15.30%	16.55%	
6	50000	24.50%	25.30%	24.90%	
7	80000	52.90%	54.70%	53.80%	
8	100000	69.00%	68.80%	68.90%	
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27