第四届（2019）全国高校密码数学挑战赛

赛题三

一、赛题名称：加法链问题

二、赛题描述：

2.1 符号说明

用符号n表示正整数，$h(n)$表示n的二进制表示中1的个数，$\lambda(n)$表示n的以2为底的对数（如果不是整数则向下取整），$\ell(n)$表示可计算n的最短加法链长度.

2.2 基础知识

给定一个正整数n，一个长度为r的可计算n的加法链(addition chain) U是一个严格递增的正整数序列$U=(u_0,u_1,u_2,\cdots,u_r)$，其中$u_0=1,u_1=2,\cdots,u_r=n$，且对任意的$k>1$，$u_k$是它前面两个元素(不必不同)的和，即存在$i,j<k$使得$u_k=u_i+u_j$. 例如，一个长为6的可计算23的加法链如下：1,2,4,5,9,18,23.可计算n的加法链不是唯一的，诸如可计算170的加法链有：1-2-4-8-12-16-18-34-68-102-170，1-2-4-8-16-18-34-68-136-170等等. 当然，找到的可计算n的加法链长度越短越好，但对于给定的n，找一个具有最短加法链长度的加法链是很困难的问题.

2.3 问题描述

加法链问题(ACP)：给定正整数n(具体的数值请参见附件:加法链问题数据文件.txt)，设$n-5\le m\le n+5$，指出哪一个m值具有最短的加法链表示，并给出其加法链表示. 加法链打印规则如下：设$U=(u_0,u_1,u_2,\cdots,u_r)$是可计算$n$的加法链，则按照如下规则进行打印：

$n=n,\ell=r,v_0=1,v_1=2,\cdots,v_k=(u_k,k,i,j),\cdots$其中$u_k=u_i+u_j,k\ge 2$，例如

$n=23,\ell=6,v_0=1,v_1=2,\ v_2=(4,2,1,1),v_3=(5,3,0,1),v_4=(9,4,2,3),$

$v_5=(18,5,4,4),v_6=(23,6,3,5)$

2.4 成绩评判

本赛题共分七类挑战问题，

1).第一至第七类挑战问题的分值分别为 5,10,15,25,35,50,60,总分值 200 分. 分数相同的选手依照难度最高的挑战问题求解时间来排序，求解用时越少者排名越靠前；

2). 针对每类挑战，给出计算平台和计算结果，并简述求解原理、步骤和实现效率（包括计算需要的时间和空间等）；

3). 如果你不能找到具有最短加法链长度的m及其加法链表示，可以给出问题的近似解，即在所要求的范围内找到某个m，使其加法链表示尽可能的短，但分数会扣减。注意，如果你给出的加法链表示是接近平凡的加法链表示(当然它肯定不是问题的正确解)，则不能得分；

4). 利用特殊算法求解或求解算法中有创新内容的，酌情加分；

5). 参赛者在报告摘要中明确列出每类问题的解：具有最短加法链长度的数和其加法链长度. 在报告正文中列出具有最短加法链长度的数及其加法链表示(按照要求打印加法链表示，否则不计分)，详细描述每个问题的求解方法。引用前人的方法需在解题报告中明确指出，否则乘积作废。

三、密码学背景及相关问题的研究进展

模指数的幂运算是公钥密码学中的核心运算之一，其运行效率直接影响着公钥密码体制的执行速度，加法链则能应用到模指数的幂运算中。同时，加法链也被应用到椭圆曲线密码中改进点乘运算的效率。关于加法链问题的历史和发展请参考文献[1,2]。加法链相关问题的研究是密码学等相关研究领域中的热门问题，研究文献很多，可参考[2-4]. $\ell(n)$表示可计算n的最短加法链长度，则有

$\ell(n)\le\lambda(n)+h(n)-1$.一个公开的猜想是$\ell(n)$的下界趋近于$\lambda(n)+\log_2(h(n))$.截止到2016年底，Neill Clift 计算出了所有小于2^{36}的正整数的最短加法链，可参看 [4]. 2018年7月，Neill Clift 对任意的$\ell(n)\le 8$，证明了

$\ell(2^n-1)=n+\ell(n)-1$.

四、参考文献

[1] 高德纳（Donald E.Knuth）著，计算机程序设计艺术(卷 2)半数值算法（第 3 版），人民邮电出版社, 2016-07-01.
[2] Noma, Adamu Muhammad , et al. A Review on Heuristics for Addition Chain Problem: Towards Efficient Public Key Cryptosystems. Journal of Computer Science 13.8(2017): 275-289.
[3] N.M. Clift, Calculating optimal addition chains, Computing V.91 , 2011, pp. 265-284.
[4] http://wwwhomes.uni-bielefeld.de/achim/addition_chain.html