南京信息工程大学 实验（实习）报告

实验名称 SHA-1算法的实现与分析 实验（实习）日期 2023/11/15 指导教师 高光勇

系 计算机 专业 奇安信 年级 2021 班次 1 姓名 朱宸扬 学号 202183760012

一、实验目的

熟悉SHA-1算法的运行过程，能够编写实现SHA-1算法程序，增加对摘要函数的理解。

二、实验要求

1 理解SHA-1轮函数的定义和工作过程。

2 实现SHA-1算法。

3 分析SHA-1算法运行的性能

1. 实验内容 (要求写出算法原理、实验核心代码及运行结果)

第一步，补齐，最后一个512，模512为448，最后64bit为长度

第二步，每个512分成16\*32bit，算出w[0]~w[80]

第三步，初始化五个寄存器，经过20\*4次轮换(a与w[j]有关)，最后的寄存器组合在一起

import struct  
  
def left\_rotate(n, b):  
 return ((n << b) | (n >> (32 - b))) & 0xFFFFFFFF  
  
def sha1(message):  
 # Initialize variables  
 h0 = 0x67452301  
 h1 = 0xEFCDAB89  
 h2 = 0x98BADCFE  
 h3 = 0x10325476  
 h4 = 0xC3D2E1F0  
  
 # Pre-processing  
 original\_message = message  
 message = bytearray(message, 'utf-8')  
 ml = 8 \* len(message)  
 message.append(0x80)  
 while len(message) % 64 != 56:  
 message.append(0x00)  
 message += struct.pack('>Q', ml)  
  
 # Process the message in 512-bit chunks  
 for i in range(0, len(message), 64):  
 chunk = message[i:i+64]  
  
  
 w = [0] \* 80  
 for j in range(16):  
 w[j] = struct.unpack('>I', chunk[j \* 4:j \* 4 + 4])[0]  
  
 for j in range(16, 80):  
 w[j] = left\_rotate(w[j-3] ^ w[j-8] ^ w[j-14] ^ w[j-16], 1)  
  
 a, b, c, d, e = h0, h1, h2, h3, h4  
  
 for j in range(80):  
 if 0 <= j <= 19:  
 f = (b & c) | ((~b) & d)  
 k = 0x5A827999  
 elif 20 <= j <= 39:  
 f = b ^ c ^ d  
 k = 0x6ED9EBA1  
 elif 40 <= j <= 59:  
 f = (b & c) | (b & d) | (c & d)  
 k = 0x8F1BBCDC  
 elif 60 <= j <= 79:  
 f = b ^ c ^ d  
 k = 0xCA62C1D6  
  
 temp = (left\_rotate(a, 5) + f + e + k + w[j]) & 0xFFFFFFFF  
 e = d  
 d = c  
 c = left\_rotate(b, 30)  
 b = a  
 a = temp  
  
 h0 = (h0 + a) & 0xFFFFFFFF  
 h1 = (h1 + b) & 0xFFFFFFFF  
 h2 = (h2 + c) & 0xFFFFFFFF  
 h3 = (h3 + d) & 0xFFFFFFFF  
 h4 = (h4 + e) & 0xFFFFFFFF  
  
 # Produce the final hash value  
 digest = f"{h0:08x}{h1:08x}{h2:08x}{h3:08x}{h4:08x}"  
 return digest  
  
# Example usage  
data\_to\_hash = "Hello, World!"  
sha1\_result = sha1(data\_to\_hash)  
print(f"SHA-1 hash of '{data\_to\_hash}': {sha1\_result}")

SHA-1 hash of 'Hello, World!': 0a0a9f2a6772942557ab5355d76af442f8f65e01

四、实验心得

熟悉了Sha-1算法