实验报告4 SHA-1实现

> 17341125　蒙亚愿　信息安全

一、实验目的

> 了解SHA-1算法的基本原理，掌握SHA-1算法的实现方法，完成字符串的SHA-1运算及算法流程。

二、实验要求

> 输入一个不小于2KB的字符串，输出其加密后的散列值。

三、实验介绍

3.1 实验综述

SHA-1的加密过程可以根据课本所给的伪代码参考实现。首先，是对字符串数据分组，分成每$512bit$（64字节）一组，通过$Hash$函数的通用结构，对每一组计算后的结果都会用于下一组，即每一组的输入值是该组字符串（$512bit$)数据和上一组计算的结果($160bit$)，输出作为下一组的输入之一。

在每一组的计算中，会有四次的迭代计算，每次迭代都要经过$20steps$，在每一个$steps$中，都会把上一组计算的结果分组，每$32bit$一组，对于这5个字，进行基本的运算，其中每一个$steps$都会有一些不同的变量参与计算。总而言之，对于每一个分组都要进行80次循环计算，然后传递给下一组，最后一组的计算结果即为SHA-1的计算结果。

3.2 开发环境

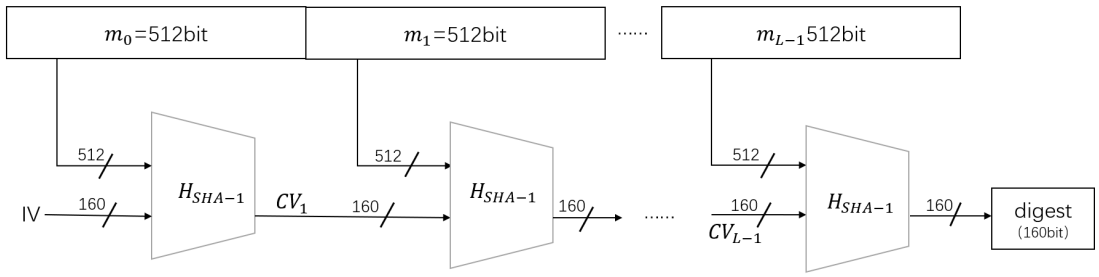
运行环境：Windows

使用语言：C++

四、算法原理及代码实现

首先是$Hash$函数的通用结构。我们可以看到，对于字符串数据，我们需要先把他们分组，分成每$512bit$也就是64字节一组，对于每一组都进行计算，计算的得到的结果（$160bit$）用于下一组的计算。其中，在第一组的使用的初始变量是规定好的，IV分成4个字：

```
h[0] = 0x67452301 ;
h[1] = 0xefcdab89 ;
h[2] = 0x98badcfe ;
h[3] = 0x10325476 ;
h[4] = 0xc3d2e1f0 ;
```



H_{SHA-1}函数由三个重要的部分组成：填充函数$padding()$、初始化变量函数$initial()$以及计算散列值的函数$cal()$。

```
void sha_1(char m[3072], unsigned long mac[5]){
    unsigned long meg[3072] ;

    int len = padding(m, meg) ;
    initial() ;  //初始化h k
    cal(meg, len, mac) ;

}
```

4.1 填充函数padding()

输入：字符串数据m，转换成$unsigned\ long$类型的数组meg

输出：转换类型之后的数组长度

首先是转换类型，把$char$类型的数组转换成$unsigned\ long$类型，根据字符的ASCII码直接转换。

接着是填充，如果字符串本来的长度l已经满足$l = 56 mod 64$，也就是l的比特位模512等于448，就不用再进行填充。否则，填充的比特位中，第一位就是1，剩下全都补充0，直到补充后的比特位模512等于448为止，也就是首先补充的第一位是十六进制的$0x80$，剩下的都是0，直到l满足$l = 56 mod 64$为止。

```
int padding(char m[3072], unsigned long meg[3072]){
    int len = strlen(m) ;
    for(int i=0;i<len;i++)  meg[i] = m[i] ;

    //  每一字母（数字）化成二进制的8位
    int i=1;
    if(len%64!=56){
        meg[len] = 0x80;
        for(i;(i+len)%64!=56;i++){
            meg[i+len]=0x0 ;
        }
    }

    // 最后的64bit用于放数据长度  64/8=8；也就是8字节
    bitset<64> temp(len*8) ;
    string lenPad = temp.to_string() ;
    int j=0;
    for(j;j<8;j++){
```

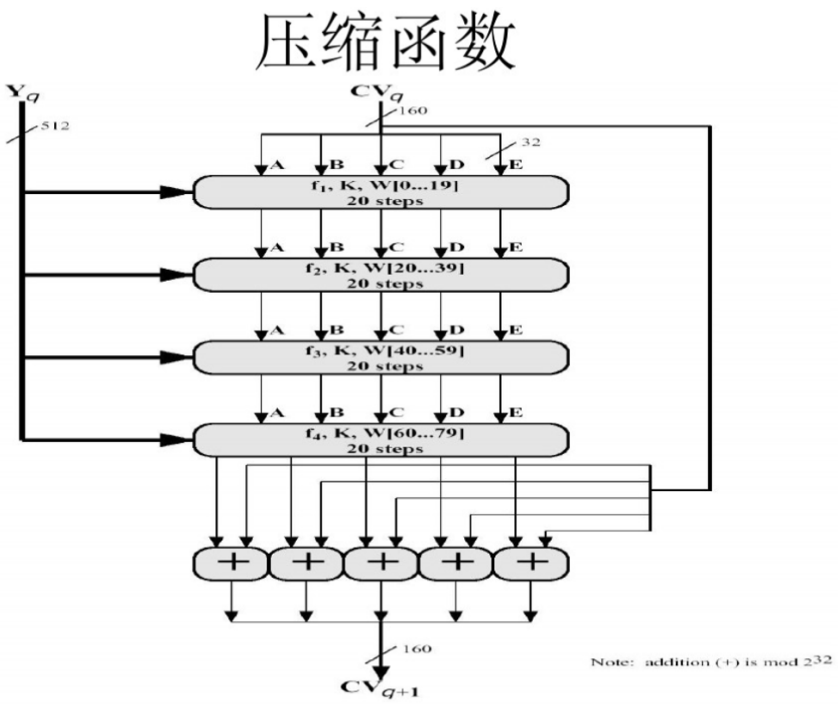
```
        bitset<8> t(lenPad,j*8,8) ;
        meg[len+i+j] = t.to_ulong() ;
    }

    len = len+i+j;
    return len ;
}
```

最后加入字符串数据的长度，放在最后的64比特，也就是4字节。如果不够存放，则取低64位。数据是大端存储。所以我首先把字符串长度化成长为64的$bitset$类型，每8位的进行转换，放入meg数组中。

4.2初始化变量函数initial()

下图为H_{SHA-1}的结构：



在四次迭代中，需要变量CV_q、K以及W，除了W需要根据字符串段Y_q来决定，其他都是 可以先初始化的，规定如下：

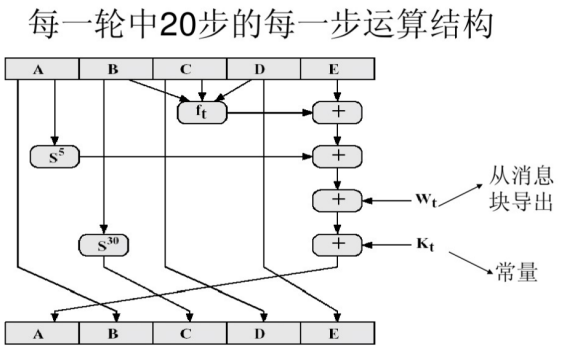
```
void initial(){
    h[0] = 0x67452301 ;
    h[1] = 0xefcdab89 ;
    h[2] = 0x98badcfe ;
    h[3] = 0x10325476 ;
    h[4] = 0xc3d2e1f0 ;

    k[0] = 0x5a827999 ;
    k[1] = 0x6ed9eba1 ;
    k[2] = 0x8f1bbcdc ;
    k[3] = 0xca62c1d6 ;
}
```

在之后的计算中，h会随着迭代的计算而发生变化。而k不会，在其中的20$steps$中，都会用的k。

4.3 计算散列值函数cal()

下图为每个$steps$中的结构：



在每一$step$中，对于输入变量CV_i，也是进行分组，分成5个字A, B, C, D, E，这五个字进行相应的运算之后得到新的五个字，继续进行下一个$step$的计算。四轮总共有80个$steps$。

4.3.1 变量w的计算

在进行计算中，需要用到变量w，w与分组后的字有关：前16个$w[t]$等于分组后的字，之后的17到79个则是前面的字的运算结果。

由于一个$w[t]$是32位，而分组之后的$block[i]$是7位，所以前16个在赋值的时候，$block[i]\,|\,block[i+1]\,|\,block[i+2]\,|\,block[i+3]$才能组成一个$w[t]$。

之后的$w[t]$则等于$w[t-3] \bigoplus w[t-8] \bigoplus w[t-14] \bigoplus w[t-16]$。

```
void cal_w(unsigned long block[64]){  //64*8 = 512bit
    //每4个block(8位)组成一个w(32位)
    int cnt=0;
    for(int i=0;i<64;i+=4){
        bitset<32> t1(block[i]) , t2(block[i+1]), t3(block[i+2]), t4(block[i+3])
;
        t1 = t1<<24;
        t2 = t2<<16;
        t3 = t3<<8 ;
        bitset<32> ans = t1^t2^t3^t4;
```

```
        w[cnt++] = ans.to_ulong() ;
    }

    for(int i=16;i<80;i++){
        unsigned long t = w[i-3]^w[i-8]^w[i-14]^w[i-16] ;
        w[i] = rotl(t,1) ;
    }

}
```

4.3.2 变量k的使用

规定如下：

轮数t	k_t
0 ≤ t ≤ 19	0x5a827999
20 ≤ t ≤ 39	0x6ed9eba1
40 ≤ t ≤ 59	0x8f1bbcdc
60 ≤ t ≤ 79	0xca62c1d6

也就是声明大小为4的$unsigned\ \ long$数组，初始化k，当$t/2$等于0时，$k[t]=0x5a827999$；等于1时，$k[t]=0x6ed9eba1$；等于2时，$k[t]=0x8f1bbcdc$；等于3时，$k[t]=0xca62c1d6$。

```
    k[0] = 0x5a827999 ;
    k[1] = 0x6ed9eba1 ;
    k[2] = 0x8f1bbcdc ;
    k[3] = 0xca62c1d6 ;
```

4.3.3 f()函数

输入：$unsigned\ \ long$类型的B,C,D

输出：$unsigned\ \ long$类型的计算结果

f()在不同的轮数也有不用的算法：

轮数t	f()
0 ≤ t ≤ 19	(B⋀C) ⋁ (~B⋀D)
20 ≤ t ≤ 39	B⨁C⨁D
40 ≤ t ≤ 59	(B⋀C)⋁(B⋀D)⋁(C⋀D)
60 ≤ t ≤ 79	B⨁C⨁D

同理可得：

```
unsigned long f(unsigned long t, unsigned long b, unsigned long c, unsigned long
d){
    if(t/20==0){
        return (b&c) | (~b&d) ;
    }
    if(t/20==2){
        return (b&c) | (b&d) | (c&d) ;
    }
    return b^c^d ;
}
```

4.3.4 rolt移位函数

输入：$unsigned\ long$类型的变量x，int类型的$shift$

输出：x循环左移$shift$位

```
unsigned long rotl(unsigned long x, int shift){
    return (x<<shift)|(x>>(32-shift)) ;
}
```

4.3.5 cal()函数

输入：$unsigned\ long$类型的字符串数据，填充后的长度len，返回的散列值mac

此函数结合以上的图以及课本上的伪代码，把各个小函数的计算结果总和即可。

每一个分组计算完之后，需更新h，以作为下一个分组计算的输入。

最后得到的h即为散列值，返回。

```
void cal(unsigned long meg[3072], int len, unsigned long mac[5]){
    unsigned long block[64] ;  //64 *8 = 512bit
    int blockCnt = len/64 ;
    int cnt=0;

    for(int i=0;i<blockCnt;i++){
        for(int j=0;j<64;j++){
            block[j] = meg[cnt++] ;
        }

        cal_w(block);

        unsigned long a=h[0], b=h[1], c=h[2], d=h[3], e=h[4] ;

        for(int t=0;t<80;t++){
            unsigned long fans = f(t,b,c,d) ;
            unsigned long temp = rotl(a,5) + fans + e + w[t] + k[t/20] ;
            e = d ;
            d = c ;
            c = rotl(b,30) ;
            b = a ;
            a = temp ;
        }
```

```
        h[0] += a ;
        h[1] += b ;
        h[2] += c ;
        h[3] += d ;
        h[4] += e ;

    }

    mac[0] = h[0] ;
    mac[1] = h[1] ;
    mac[2] = h[2] ;
    mac[3] = h[3] ;
    mac[4] = h[4] ;
}
```

其中的加法是在模2^{32}上运算的，目的是结果不超过$32bit$，如果超过的话只取低32位。但是使用$unsigned\ long$类型的话直接加就可以了。

4.4 main函数

此函数主要用于输入字符串数据，返回散列值。

```
int main(){
    char meg[3072];
    unsigned long mac[5];

    while(1){
        cout<<"请输入SHA-1加密字符串：\n";
        gets(meg) ;
        sha_1(meg, mac);
        printf("\n散列值：
%08x%08x%08x%08x%08x\n\n\n\n",mac[0],mac[1],mac[2],mac[3],mac[4]) ;
    }
//  system("pause") ;
}
```

五、结果测试

测试一

输入：
Aaitacameafromainatimeatoaseeaaayoungawomanalookingadown."Isathisayours?"aheaasked.Sheasaid,a"Yes,acouldayouabringaitaup?"aandatheamanaagreed.Onaarrivalasheawasaprofuseainaherathanksaandaofferedatheamanaaadrink.aAsasheawasaveryaattractiveaheaagreed.aShortlyaafterwardsasheasaid,a"I'maaboutatoahaveadinner.aThere'saplenty.aWouldayoualikeatoajoiname?"Heareadilyaacceptedaheraofferaandabothaenjoyedaaalovelyameal.aAsatheaeveningawasadrawingatoaaacloseathealadyasaid,a"I'veahadaaamarvelousaevening.aWouldayoualikeatoastayatheanight?"Theamanahesitatedathenasaid,a"Doayouaactalikeathisawithaeveryamanayouameet?""No,"ashereplied,a"Onlyathoseawhoacatchamyaeye."aTRUMP:aChiefaJusticeaRoberts,aPresidentaCarter,aPresidentaClinton,aPresidentaBush,aPresidentaObama,afellowaAmericansaandapeopleaofat

heaworld,athankayou.We,atheacitizensaofaAmerica,aareanowajoinedainaaagreatanationalaeffortatoarebuildaouracountryaandarestoreaitsapromiseaforaallaofaourapeople.Together,aweawilladetermineatheacourseaofaAmericaaandatheaworldaforamany,amanyayearsatoacome.aWeawillafaceachallenges,aweawillaconfrontahardships,abutaweawillagetatheajobadone.Everyafourayears,aweagatheraonatheseastepsatoacarryaoutatheaorderlyaandapeacefulatransferaofapower,aandaweaareagratefulatoaPresidentaObamaaandaFirstaLadyaMichelleaObamaaforatheiragraciousaaidathroughoutathisatransition.aTheyahaveabeenamagnificent.aThankayou.Todaysacapitalahasareapedathearewardsaofagovernmentawhileatheapeopleahaveaborneatheacost.aWashingtonaflourished,abutatheapeopleadidanotashareainaitsawealth.aPoliticiansaprospered,abutatheajobsaleftaandatheafactoriesaclosed.aTheaestablishmentaprotectedaitself,abutanotatheacitizensaofaouracountry.aTheiravictoriesahaveanotabeenayouravictories.aTheiratriumphsahaveanotabeenayouratriumphs.aAndawhileatheyacelebratedainaouranationveaenrichedaforeignaindustryaatatheaexpenseaofaAmericanaindustry;asubsidizedatheaarmiesaofaotheracountries,awhileaallowingaforatheaveryasadadepletionaofaouramilitary.aWeabordersawhilearefusingatoadefendaouraown.Andaspentatrillionsaandatrillionsaofadollarsaoverseasawhileaamericaveamadeaotheracountriesarich,awhileatheawealth,astrengthaandaconfidenceaofaouracountryahasadissipate

输出：

47c42f2514f1e865508762a54958cbef7cbb772e

（20个字节，共160bit）

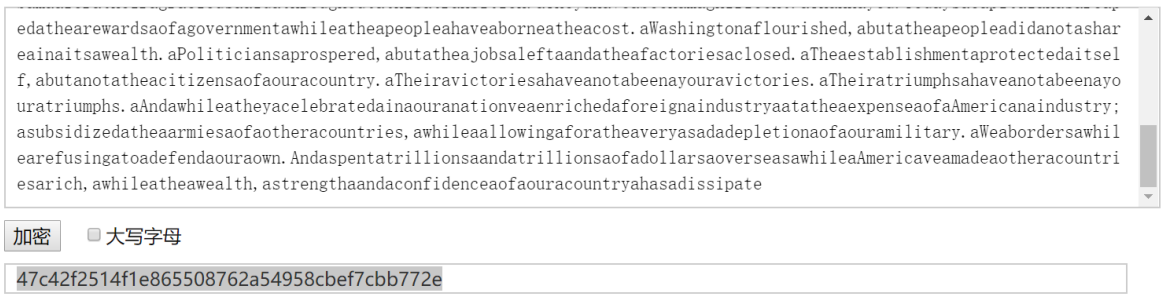
```
ips,abutaweawillagetatheajobadone.Everyafourayears,aweagatheraonatheseastepsatoacarryaoutatheaorder
lyaandapeacefulatransferaofapower,aandaweaareagratefulatoaPresidentaObamaaandaFirstaLadyaMichelleaO
bamaaforatheiragraciousaaidathroughoutathisatransition.aTheyahaveabeenamagnificent.aThankayou.Today
sacapitalahasareapedathearewardsaofagovernmentawhileatheapeopleahaveaborneatheacost.aWashingtonaflo
urished,abutatheapeopleadidanotashareainaitsawealth.aPoliticiansaprospered,abutatheajobsaleftaandat
heafactoriesaclosed.aTheaestablishmentaprotectedaitself,abutanotatheacitizensaofaouracountry.aTheir
avictoriesahaveanotabeenayouravictories.aTheiratriumphsahaveanotabeenayouratriumphs.aAndawhileathey
acelebratedainaouranationveaenrichedaforeignaindustryaatatheaexpenseaofaAmericanaindustry;asubsidiz
edatheaarmiesaofaotheracountries,awhileaallowingaforatheaveryasadadepletionaofaouramilitary.aWeabor
dersawhilearefusingatoadefendaouraown.AndaspentatrillionsaandatrillionsaofadollarsaoverseasawhileaA
mericaveamadeaotheracountriesarich,awhileatheawealth,astrengthaandaconfidenceaofaouracountryahasadi
ssipate

散列值：47c42f2514f1e865508762a54958cbef7cbb772e
```

网站验证：https://www.qqxiuzi.cn/bianma/sha-1.htm

47c42f2514f1e865508762a54958cbef7cbb772e

结果一样，答案正确。



测试二

输入：

eaitacameafromainatimeatoaseeaaayoungawomanalookingadown."Isathisayours?"aheaasked.Sheasaid,a"Yes,acouldayouabringaitaup?"aandatheamanaagreed.Onaarrivalasheawasaprofuseainaherathanksaandaofferedatheamanaaadrink.aAsasheawasaveryaattractiveaheaagreed.aShortlyaafterwardsasheasaid,a"I'maaboutatoahaveadinner.aThere'saplenty.aWouldayoualikeatoajoiname?"Heareadilyaacceptedaheraofferaandabothaenjoyedaaalovelyameal.aAsatheaeveningawasadrawingatoaaacloseathealadyasaid,a"I'veahadaaamarvelousaevening.aWouldayoualikeatoastayatheanight?"Theamanahesitatedathenasaid,a"Doayouaactalikeathisawithaeveryamanayouameet?""No,"asheareplied,a"Onlyathoseawhoacatchamyaeye."aTRUMP:aChiefaJusticeaRoberts,aPresidentaCarter,aPresidentaClinton,aPresidentaBush,aPresidentaObama,afellowaAmericansaandapeopleaofatheaworld,athankayou.We,atheacitizensaofaAmerica,aareanowajoinedainaaagreatanationalaeffortatoarebuildaouracountryaandarestoreaitsapromiseaforaallaofaourapeople.Together,aweawilladetermineatheacourseaofaAmericaaandatheaworldaforamany,amanyayearsatoacome.aWeawillafaceachallenges,aweawillaconfrontahardships,abutaweawillagetatheajobadone.Everyafourayears,aweagatheraonatheseastepsatoacarryaoutatheaorderlyaandapeacefulatransferaofapower,aandaweaareagratefulatoaPresidentaObamaaandaFirstaLadyaMichelleaObamaaforatheiragraciousaaidathroughoutathisatransition.aTheyahaveabeenamagnificent.aThankayou.Todaysacapitalahasareapedathearewardsaofagovernmentawhileatheapeopleahaveaborneatheacost.aWashingtonaflourished,abutatheapeopleadidanotashareainaitsawealth.aPoliticiansaprospered,abutatheajobsaleftaandatheafactoriesaclosed.aTheaestablishmentaprotectedaitself,abutanotatheacitizensaofaouracountry.aTheiravictoriesahaveanotabeenayouravictories.aTheiratriumphsahaveanotabeenayouratriumphs.aAndawhileatheyacelebratedainaouranationveaenrichedaforeignaindustryaatatheaexpenseaofaAmericanaindustry;asubsidizedatheaarmiesaofaotheracountries,awhileaallowingaforatheaveryasadadepletionaofaouramilitary.aWeabordersawhilearefusingatoadefendaouraown.AndaspentatrillionsaandatrillionsaofadollarsaoverseasawhileaAmericaveamadeaotheracountriesarich,awhileatheawealth,astrengthaandaconfidenceaofaouracountryahasadissipate

输出：

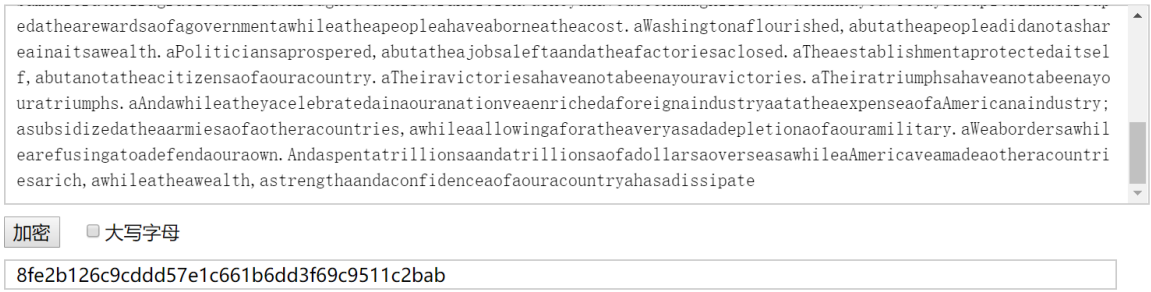
8fe2b126c9cddd57e1c661b6dd3f69c9511c2bab（改变第一个字母，达到雪峰效应）

```
rld,athankayou.We,atheacitizensaofaAmerica,aareanowajoinedainaaagreatanationalaeffortatoarebuildaou
racountryaandarestoreaitsapromiseaforaallaofaourapeople.Together,aweawilladetermineatheacourseaofaA
mericaaandatheaworldaforamany,amanyayearsatoacome.aWeawillafaceachallenges,aweawillaconfrontahardsh
ips,abutaweawillagetatheajobadone.Everyafourayears,aweagatheraonatheseastepsatoacarryaoutatheaorder
lyaandapeacefulatransferaofapower,aandaweaareagratefulatoaPresidentaObamaaandaFirstaLadyaMichelleaO
bamaaforatheiragraciousaaidathroughoutathisatransition.aTheyahaveabeenamagnificent.aThankayou.Today
sacapitalahasareapedathearewardsaofagovernmentawhileatheapeopleahaveaborneatheacost.aWashingtonaflo
urished,abutatheapeopleadidanotashareainaitsawealth.aPoliticiansaprospered,abutatheajobsaleftaandat
heafactoriesaclosed.aTheaestablishmentaprotectedaitself,abutanotatheacitizensaofaouracountry.aTheir
avictoriesahaveanotabeenayouravictories.aTheiratriumphsahaveanotabeenayouratriumphs.aAndawhileathey
acelebratedainaouranationveaenrichedaforeignaindustryaatatheaexpenseaofaAmericanaindustry;asubsidiz
edatheaarmiesaofaotheracountries,awhileaallowingaforatheaveryasadadepletionaofaouramilitary.aWeabor
dersawhilearefusingatoadefendaouraown.AndaspentatrillionsaandatrillionsaofadollarsaoverseasawhileaA
mericaveamadeaotheracountriesarich,awhileatheawealth,astrengthaandaconfidenceaofaouracountryahasadi
ssipate

散列值：8fe2b126c9cddd57e1c661b6dd3f69c9511c2bab
```

网站验证：8fe2b126c9cddd57e1c661b6dd3f69c9511c2bab

结果一样，答案正确。



六、实验总结

- 本次实验比较容易，课本里有很详细的参考伪代码，照着打就可以。只是分组那一块比较疑惑，我看到有的博客说当长度刚刚好模512等于448的时候，也要在后面补充一个比特1，也就是不管怎么先在最后一位补上1，补位的长度就是0~512，但实际上好像不是这样？我实现的是不足的话不用填充。
- 用分组块来初始化w的时候有点绕，没有注意到每个w都是一个字，所以一开始实现的是逐位赋值，就是$w[i] = block[i]$，然后答案一直都不对，后来改过来一下就对了……下次要更加细心一点。
- 还有一点就是我记得SHA-1的数据存储模式是大端存储，低位在高地址，也就是说在填充的后面只有$64bit$用于存放字符串数据长度，当64位不够用时，取低位。这一点的实现也是试了一下正好对ahhh，所以感觉如果是MD5的话，我可能就得再多花点心思想想了emmm
- 最后辛苦老师和ta啦~