Shivaprasad B J
210948023

# AC LAB

## Week 5

Write a C program to implement MD5 hashing technique.

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
#include<string.h>

typedef union uwb
{
unsigned w;
unsigned char b[4];
}MD5union;

typedef unsigned DigestArray[4];

unsigned func0(unsigned abcd[])
{
return(abcd[1] & abcd[2]) | (~abcd[1] & abcd[3]);
}

unsigned func1(unsigned abcd[])
{
return(abcd[3] & abcd[1]) | (~abcd[3] & abcd[2]);
}
```

```
unsigned func2(unsigned abcd[])
{
return abcd[1] ^ abcd[2] ^ abcd[3];
}

unsigned func3(unsigned abcd[])
{
return abcd[2] ^ (abcd[1] |~ abcd[3]);
}

typedef unsigned (*DgstFctn) (unsigned a[]);

unsigned *calctable(unsigned *k)
{
double s,pwr;
int i;
pwr = pow(2,32);
for(i=0;i<64;i++)
{
s=fabs(sin(1+i));
k[i]=(unsigned)(s*pwr);
}
return k;
}
```

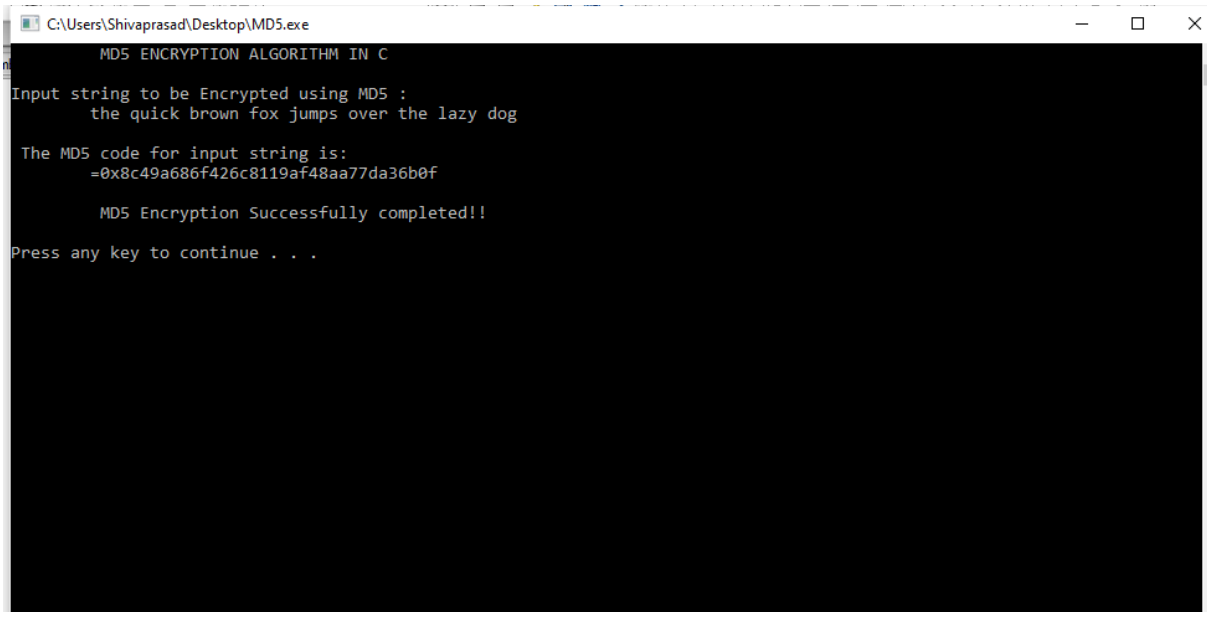
```
unsigned rol(unsigned r,short N)
{
unsigned mask1 = (1<<N)-1;
return ((r>>(32-N))& mask1) | ((r<<N) & ~mask1);
}

unsigned *md5(const char *msg, int mlen)
{
static DigestArray h0 = { 0x67452301, 0xEFCDAB89,0x98BADCFE, 0x10325476};
static DgstFctn ff[] = { &func0, &func1, &func2, &func3};
static short M[] = {1,5,3,7,0};
static short O[] = {0,1,5,0};
static short rot0[] = {7,12,17,22};
static short rot1[] = {5,9,14,20};
static short rot2[] = {4,11,16,23};
static short rot3[] = {6,10,15,21};
static short *rots[] = {rot0,rot1,rot2,rot3};
static unsigned kspace[64];
static unsigned *k;
static DigestArray h;
DigestArray abcd;
DgstFctn fctn;
short m,o,g;
unsigned f;
short *rotn;
```

```
union

{

unsigned w[16];

char b[64];

}mm;



int os = 0;

int grp,grps,q,p;

unsigned char *msg2;

if (k==NULL)

k=calctable(kspace);

for(q=0;q<4;q++) h[q]=h0[q];

  {

     grps=1+(mlen+8)/64;

     msg2=malloc(64*grps);

     memcpy(msg2,msg,mlen);

     msg2[mlen]=(unsigned char)0x80;

     q=mlen+1;

     while(q<64*grps){msg2[q]=0;q++;}

     {

       MD5union u;

       u.w=8*mlen;

       q-=8;

       memcpy(msg2+q,&u.w,4);

     }
```

```
    }


    for(grp=0;grp<grps;grp++)
    {
        memcpy(mm.b,msg2+os,64);
        for(q=0;q<4;q++) abcd[q]=h[q];


for(p=0;p<4;p++)
{
    fctn = ff[p];
    rotn = rots[p];
    m=M[p];
    o=O[p];
    for(q=0;q<16;q++)
    {
        g=(m*q+0)%16;
        f=abcd[1]+rol(abcd[0]+fctn(abcd)+k[q+16*p]+mm.w[g],rotn[q%4]);
        abcd[0]=abcd[3];
        abcd[3]=abcd[2];
        abcd[2]=abcd[1];
        abcd[1]=f;
    }}
    for(p=0;p<4;p++)
        h[p] += abcd[p];
    os += 64;
}
```

```
return h;
}


void main()
{
int j, k;
const char *msg = "the quick brown fox jumps over the lazy dog";
unsigned *d= md5(msg, strlen(msg));
MD5union u;


printf("\t MD5 ENCRYPTION ALGORITHM IN C \n\n");
printf("Input string to be Encrypted using MD5 : \n\t%s", msg);
printf("\n\n The MD5 code for input string is:/n");
printf("\t=0x");
for (j=0; j<4; j++)
{
u.w=d[j];
for(k=0;k<4;k++)
printf("%02x",u.b[k]);
}
printf("\n");
printf("\n\t MD5 Encryption Successfully completed!!\n\n");
system("pause");
}
```

OUTPUT:



```
         MD5 ENCRYPTION ALGORITHM IN C

Input string to be Encrypted using MD5 :
        the quick brown fox jumps over the lazy dog

 The MD5 code for input string is:
        =0x8c49a686f426c8119af48aa77da36b0f

         MD5 Encryption Successfully completed!!

Press any key to continue . . .
```