1.)

a.) Iterated Substitution Permutation Cipher (ECB mode)

Encryption Code

/\*

E4D12FB83A2A6C5907

1 4 5 7 3 6 2 8

\*/

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<ctype.h>

#include<time.h>

const int n=26;

int key[100];

int modifiedP[100],p[100],c[100];

int sBox[16],pBox[16],Nr,keySize;

void roundKeyMix(int l)

{

int i;

for(i=0;i<8;i++)

p[i]^=key[l+i];

return;

}

void substitute()

{

int i,k;

for(k=0;k<8;k+=4)

{

int value=0,size=4,temp=1;

//converting 4-bit streams to decimal equivalent

for(i=size-1;i>=0;i--)

{

value+=temp\*p[k+i];

temp\*=2;

}

value=sBox[value];

//storing the binary equivalent of the sBox value in plain-text

while(value>0)

{

p[k+size-1]=value%2;

value/=2;

size--;

}

while(size>0)

{

p[k+size-1]=0;

size--;

}

}

return;

}

void permute()

{

int i;

int temp[8];

for(i=0;i<8;i++)

temp[i]=p[pBox[i]];

for(i=0;i<8;i++)

pBox[i]=temp[i];

return;

}

void encrypt()

{

// passing through each rounds procedurally

int i;

for(i=1;i<=Nr;i++)

{

int start=(4\*i-3)-1;

// round key mixing

roundKeyMix(start);

if(i!=Nr)

{

substitute();

permute();

}

else

substitute();

}

// fifth round key mixing

i=5;

int start=(4\*i-3)-1;

roundKeyMix(start);

printf("\n\nEncrypted Text is: ");

for(i=0;i<8;i++)

printf("%d",p[i]);

printf("\n");

return;

}

int main()

{

char ch,temp[100];

int a,i;

srand(time(0));

FILE\*fp=fopen("1A\_key.txt","w");

printf("\n\n---------------------ITERATED SUBSTITUTION PERMUTATION CIPHER(ELECTRONIC CODE BOOK MODE)-----------------\n");

printf("\n--------------------------------------------------AT ALICE'S END-------------------------------------------\n");

printf("\nEnter number of rounds: ");

scanf("%d",&Nr);

keySize=8+Nr\*4;

printf("\nGenerating random key stream of %d bits and storing it in 1A\_key.txt...............",keySize);

for(i=0;i<24;i++)

{

key[i]=rand()%2;

fprintf(fp,"%d ",key[i]);

}

printf("\nKey stream is: ");

for(i=0;i<keySize;i++)

printf("%d",key[i]);

printf("\n\nEnter S-box substitution function: ");

scanf("%s",temp);

for(i=0;i<16;i++)

{

if(isupper(temp[i]))

sBox[i]=(int)((int)temp[i]-65)+10;

else

sBox[i]=(int)((int)temp[i]-48);

}

printf("\nEnter P-box permutation function: ");

for(i=0;i<8;i++)

{

scanf("%1d",&pBox[i]);

pBox[i]--;

}

printf("\nEnter the plain-text(8-bits) to be encrypted: ");

for(i=0;i<8;i++)

{

scanf("%1d",&p[i]);

}

encrypt();

fclose(fp);

return 0;

}

Decryption Code

/\*

E4D12FB83A2A6C5907

1 4 5 7 3 6 2 8

\*/

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<ctype.h>

#include<time.h>

const int n=26;

int key[100];

int modifiedP[100],p[100],c[100];

int sBox[16],pBox[8],Nr,keySize;

void roundKeyMix(int l)

{

int i;

for(i=0;i<8;i++)

c[i]^=key[l+i];

return;

}

void substitute()

{

int i,k;

for(k=0;k<8;k+=4)

{

int value=0,size=4,temp=1;

//converting 4-bit streams to decimal equivalent

for(i=size-1;i>=0;i--)

{

value+=temp\*c[k+i];

temp\*=2;

}

value=sBox[value];

//storing the binary equivalent of the sBox value in plain-text

while(value>0)

{

c[k+size-1]=value%2;

value/=2;

size--;

}

while(size>0)

{

c[k+size-1]=0;

size--;

}

}

return;

}

void permute()

{

int i;

int temp[8];

for(i=0;i<8;i++)

temp[i]=c[pBox[i]];

for(i=0;i<8;i++)

pBox[i]=temp[i];

return;

}

void invertSbox()

{

int i,temp[16];

for(i=0;i<16;i++)

temp[sBox[i]]=i;

for(i=0;i<16;i++)

sBox[i]=temp[i];

}

void invertPbox()

{

int i,temp[8];

for(i=0;i<8;i++)

temp[pBox[i]]=i;

for(i=0;i<8;i++)

pBox[i]=temp[i];

}

void decrypt()

{

//inverting S box substitution function

invertSbox();

//inverting P box substitution function

invertPbox();

// fifth round key mixing equivalent

int i=5;

int start=(4\*i-3)-1;

roundKeyMix(start);

// passing through each rounds starting from backwards procedurally

for(i=Nr;i>=1;i--)

{

int start=(4\*i-3)-1;

// round key mixing

if(i!=Nr)

{

permute();

substitute();

}

else

substitute();

roundKeyMix(start);

}

printf("\n\nDecrypted Text is: ");

for(i=0;i<8;i++)

printf("%d",c[i]);

printf("\n");

return;

}

int main()

{

char ch,temp[100];

int a,i;

srand(time(0));

FILE\*fp=fopen("1A\_key.txt","r");

printf("\n\n---------------------ITERATED SUBSTITUTION PERMUTATION CIPHER(ELECTRONIC CODE BOOK MODE)-----------------\n");

printf("\n--------------------------------------------------AT BOB'S END-------------------------------------------\n");

printf("\nEnter number of rounds: ");

scanf("%d",&Nr);

keySize=8+Nr\*4;

for(i=0;i<keySize;i++)

fscanf(fp,"%d",&key[i]);

printf("\nKey stream is: ");

for(i=0;i<keySize;i++)

printf("%d",key[i]);

printf("\n\nEnter S-box substitution function: ");

scanf("%s",temp);

for(i=0;i<16;i++)

{

if(isupper(temp[i]))

sBox[i]=(int)((int)temp[i]-65)+10;

else

sBox[i]=(int)((int)temp[i]-48);

}

printf("\nEnter P-box permutation function: ");

for(i=0;i<8;i++)

{

scanf("%1d",&pBox[i]);

pBox[i]--;

}

printf("\nEnter the cipher text(8-bits) to be decrypted: ");

for(i=0;i<8;i++)

{

scanf("%1d",&c[i]);

}

decrypt();

fclose(fp);

return 0;

}

Encryption Output



Decryption Output

