#include<stdio.h>

#include<math.h>

int gcd(int a, int h)

{

int temp;

while(1)

{

temp = a%h;

if(temp==0)

return h;

a = h;

h = temp;

}

}

int main()

{

double p = 3;

double q = 7;

double n=p\*q;

double count;

double totient = (p-1)\*(q-1);

double e=2;

while(e<totient){

count = gcd(e,totient);

if(count==1)

break;

else

e++;

}

double d;

double k = 2;

d = (1 + (k\*totient))/e;

double msg = 12;

double c = pow(msg,e);

double m = pow(c,d);

c=fmod(c,n);

m=fmod(m,n);

printf("Message data = %lf",msg);

printf("\np = %lf",p);

printf("\nq = %lf",q);

printf("\nn = pq = %lf",n);

printf("\ntotient = %lf",totient);

printf("\ne = %lf",e);

printf("\nd = %lf",d);

printf("\nEncrypted data = %lf",c);

printf("\nOriginal Message Sent = %lf",m);

return 0;

}

Serve:=

#include <sys/socket.h>

#include <netinet/in.h>

#include <arpa/inet.h>

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <errno.h>

#include <string.h>

#include <sys/types.h>

int main(void)

{

int listenfd = 0,connfd = 0;

struct sockaddr\_in serv\_addr;

char sendBuff[1025];

int numrv;

listenfd = socket(AF\_INET, SOCK\_STREAM, 0);

printf("socket retrieve success\n");

memset(&serv\_addr, '0', sizeof(serv\_addr));

memset(sendBuff, '0', sizeof(sendBuff));

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_addr.s\_addr = htonl(INADDR\_ANY);

serv\_addr.sin\_port = htons(5000);

bind(listenfd, (struct sockaddr\*)&serv\_addr,sizeof(serv\_addr));

if(listen(listenfd, 10) == -1){

printf("Failed to listen\n");

return -1;

}

while(1)

{

connfd = accept(listenfd, (struct sockaddr\*)NULL ,NULL); // accept awaiting request

strcpy(sendBuff, "Message from server");

write(connfd, sendBuff, strlen(sendBuff));

close(connfd);

sleep(1);

}

return 0;

}

**Client**

#include <sys/socket.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <netdb.h>

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <unistd.h>

#include <errno.h>

#include <arpa/inet.h>

int main(void)

{

int sockfd = 0,n = 0;

char recvBuff[1024];

struct sockaddr\_in serv\_addr;

memset(recvBuff, '0' ,sizeof(recvBuff));

if((sockfd = socket(AF\_INET, SOCK\_STREAM, 0))< 0)

{

printf("\n Error : Could not create socket \n");

return 1;

}

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_port = htons(5000);

serv\_addr.sin\_addr.s\_addr = inet\_addr("127.0.0.1");

if(connect(sockfd, (struct sockaddr \*)&serv\_addr, sizeof(serv\_addr))<0)

{

printf("\n Error : Connect Failed \n");

return 1;

}

while((n = read(sockfd, recvBuff, sizeof(recvBuff)-1)) > 0)

{

recvBuff[n] = 0;

if(fputs(recvBuff, stdout) == EOF)

{

printf("\n Error : Fputs error");

}

printf("\n");

}

if( n < 0)

{

printf("\n Read Error \n");

}

return 0;

}