TECHNO MAIN , SALT LAKE

NAME..PRAYASH KUMAR SAHA.....

ROLL NO..13000319054.....

REGISTRATION NO...040994.....

DEPT..Electronics and Communication.....

SUBJECT NAME & CODE...EC-692.....

Computer Network Lab

.....

Experiment - 8

Client.c

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#define PORT 8080
#define MAXLINE 1024
int main() {
int sockfd; char a[100],b[100];
struct sockaddr_in servaddr;
if ((sockfd=socket(AF_INET,SOCK_DGRAM,0))<0){perror("Socket Creation Failed");
exit(EXIT_FAILURE); }
memset(&servaddr, 0, sizeof(servaddr));
servaddr.sin_family = AF_INET;
servaddr.sin_port = htons(PORT);
servaddr.sin_addr.s_addr = INADDR_ANY;
printf("Enter the data : "); scanf("%s",a);
int n, len;
sendto(sockfd, (char *)a, strlen(a),
MSG_CONFIRM, (const struct sockaddr *) &servaddr,sizeof(servaddr));
perror("Send to"); n = recvfrom(sockfd, (char *)b, MAXLINE,
```

```
MSG_WAITALL, (struct sockaddr *) &servaddr,&len);
perror("recvfrom"); b[n] = '\0';
printf("Codeword receiver from Server is : %s\n", b);close(sockfd); return 0; }
```

Server.c

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#define PORT 8080
#define MAXLINE 1024
int power(int x, int y) {
int i,m = 1;
for(i = 1; i <= y; i++)
m *= x; return m; }
void HammingCode(char* a, char* b){
int n = strlen(a);
int r,m,i,j,k,count,x=1;
for(r=1; r<n; r++)
if(n+r < power(2,r)) { break; }
printf("No. of Redundant bit(s) : %d\n",r);  m = n + r;
printf("Length of the message is : %d\n",m);
for(i=1,j=0; i <= m+1; i++){
```

```
if(i == x){ x *= 2; b[i] = '0'; }
else { b[i] = a[j++]; }
for(i=1; i<=m; i*=2){ k = i; count = 0;
do{
for(j=1; j<=i && k<=m; j++,k++){
if(b[k]=='1') {count++; }
k = k+j-1;
}while(k<=m);
if(count & 1 == 1){ b[i] = '1';}
else {b[i] = '0';  }
b[m+1] = '\0'; }
int main() {
int sockfd,i,value=3;
char a[100],m[100];
m[0] = '0';
struct sockaddr_in servaddr, cliaddr;
if ( (sockfd = socket(AF_INET, SOCK_DGRAM, 0)) < 0 ) {
perror("Socket Creation failed"); // print error
exit(EXIT_FAILURE); // Clean up the module }
memset(&servaddr, 0, sizeof(servaddr));
memset(&cliaddr, 0, sizeof(cliaddr));
servaddr.sin_family = AF_INET; // IPv4
servaddr.sin_addr.s_addr = INADDR_ANY;
servaddr.sin_port = htons(PORT);
if ( bind(sockfd,(const struct sockaddr *)&servaddr,sizeof(servaddr)) < 0 ) {
perror("Bind Failed"); // print error
exit(EXIT_FAILURE);// Clean up the module  }
```

```
int len, n;
len = sizeof(cliaddr);
printf("Server is Waiting.....\n");
n = recvfrom(sockfd, (char *)a, MAXLINE,
MSG_WAITALL, ( struct sockaddr *) &cliaddr,&len);
perror("recvfrom");
printf("Received data from client: %s\n",a);
HammingCode(a,m);
printf("Codeword formed: %s\n",m);
sendto(sockfd, (char *)m, strlen(m),
MSG_CONFIRM, (const struct sockaddr *) &cliaddr,len);
return 0; }
```

OUTPUT