palindrome server :

```
#include<string.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<stdlib.h>
#include<stdio.h>

int main()
{
    int s,r,recb,sntb,x;
    int ca;
    printf("INPUT port number: ");
    scanf("%d", &x);
    socklen_t len;
    struct sockaddr_in server,client;
    char buff[50];

    s=socket(AF_INET,SOCK_DGRAM,0);
    if(s==-1)
    {
        printf("\nSocket creation error.");
        exit(0);
    }
    printf("\nSocket created.");

    server.sin_family=AF_INET;
    server.sin_port=htons(x);
    server.sin_addr.s_addr=htonl(INADDR_ANY);
    len=sizeof(client);
    ca=sizeof(client);

    r=bind(s,(struct sockaddr*)&server,sizeof(server));
    if(r==-1)
        {
        printf("\nBinding error.");
        exit(0);
    }
    printf("\nSocket binded.");
while(1){

    recb=recvfrom(s,buff,sizeof(buff),0,(struct sockaddr*)&client,&ca);
    if(recb==-1)
    {
        printf("\nMessage Recieving Failed");
        close(s);
        exit(0);
    }
    printf("\nMessage Recieved: ");
```

```
    printf("%s\n", buff);
    if(!strcmp(buff,"halt"))
        break;


    char buff2[50];
    strcpy(buff2,buff);
    buff[1]=strlen(buff2);
    int n=0;
    for(int i=0;i<buff[1];i++)
        if(buff2[i]=='a'||buff2[i]=='e'||buff2[i]=='o'||buff2[i]=='i'||buff2[i]=='u')
            n++;
    buff[2]=n;
    buff[0]=1;
    for(int i=0;i<buff[1]/2;i++)
    {
        if(buff2[i]!=buff2[buff[1]-i-1])
        {
            buff[0]=0;
            break;
        }
    }
    sntb=sendto(s,buff,sizeof(buff),0,(struct sockaddr*)&client,len);
    if(sntb==-1)
    {
        printf("\nMessage Sending Failed");
        close(s);
        exit(0);
    }

    if(!strcmp(buff,"halt"))
        break;

}
    close(s);
}
```

palindrome client :

```
#include<string.h>
#include<arpa/inet.h>
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<fcntl.h>
#include<sys/stat.h>

int main()
```

```
{
    int s,r,recb,sntb,x;
    int sa;
    socklen_t len;
    printf("INPUT port number: ");
    scanf("%d", &x);
    struct sockaddr_in server,client;
    char buff[50];
    s=socket(AF_INET,SOCK_DGRAM,0);
    if(s==-1)
    {
        printf("\nSocket creation error.");
        exit(0);
    }
    printf("\nSocket created.");
    server.sin_family=AF_INET;
    server.sin_port=htons(x);
    server.sin_addr.s_addr=inet_addr("127.0.0.1");
    sa=sizeof(server);
    len=sizeof(server);

while(1){

    printf("\n\n");
    printf("Enter new string: ");
    scanf("%s", buff);
    sntb=sendto(s,buff,sizeof(buff),0,(struct sockaddr *)&server, len);
    if(sntb==-1)
    {
        close(s);
        printf("\nMessage sending Failed");
        exit(0);
    }
    if(!strcmp(buff,"halt"))
        break;
    recb=recvfrom(s,buff,sizeof(buff),0,(struct sockaddr *)&server,&sa);
    if(recb==-1)
    {
        printf("\nMessage Recieving Failed");
        close(s);
        exit(0);
    }
    if(buff[0]==1)
    printf("\nString is palindrome! Length of string is %d and it contains %d number of vowels.
",buff[1],buff[2]);
    else
        printf("\nString is not a palindrome! Length of string is %d and it contains %d number of
vowels. ",buff[1],buff[2]);

    }
    close(s);
```

```
}
```