

PROGRAM TITLE:Develop a Client Server Application using UDP where the client will send a sentence to the server and the server will display the number of vowels, consonants and white spaces in the sentence.

PROGRAM CODE:

server.c

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

#define RECVBUFSIZE 1024
count(char recv[])
{
    int i,v=0,c=0,w=0,t;
    for(i=0;recv[i]!='\0';i++)
    {
        t=(int)recv[i];
        if((t>96&&t<123)|| (t>64&&t<91))
        {
            if(recv[i]=='a' || recv[i]=='e' || recv[i]=='i' || recv[i]=='o' ||
recv[i]=='u' || recv[i]=='A' || recv[i]=='E' || recv[i]=='I' || recv[i]=='O' || recv[i]=='U')
            {
                v++;
            }
            else
            {
                c++;
            }
        }
        else if(t==32)
            w++;
    }
    printf("\n\tThe message received is: %s\n\tNumber of vowels: %d\n\tNumber of
consonants: %d\n\tNumber of white spaces: %d\n",recv,v,c,w);
}
main()
{
    int servSock, recvBufSize,i;
    struct sockaddr_in serverAddr;
    struct sockaddr_storage serverStorage;
    socklen_t addr_size;
    char server_ip[] = "127.0.0.1";
    unsigned short server_port=25051;
    char recvBuf[RECVBUFSIZE],sendBuf[RECVBUFSIZE];
    bzero(&serverAddr,sizeof(serverAddr));
    serverAddr.sin_family = AF_INET;//Internet Address family
    serverAddr.sin_port = htons(server_port);//Local Port address
    inet_aton(server_ip,(&serverAddr.sin_addr));
    if((servSock=socket(AF_INET,SOCK_DGRAM,0))<0)//DGRAM as UDP uses DGRAM
    {
        printf("\n\tSocket Error.\n");
        exit(1);
    }
    printf("\n\tSERVER: Socket Created.\n");
    if((bind(servSock,(struct sockaddr*)&serverAddr, sizeof(serverAddr)))<0)//-1
indicates failure
```

```

    {
        printf("\n\tBind Error.\n");
        close(servSock); //Closing the socket
        exit(1);
    }
    printf("\n\tSERVER: Binded Successfully.\n");
    addr_size = sizeof(serverStorage);
    if(recvBufSize=recvfrom(servSock,recvBuf,RECVBUFSIZE,0,(struct sockaddr
*)&serverStorage, &addr_size)<0)
    {
        printf("\n\tReceive Error.\n");
    }
    count(recvBuf);
    close(servSock);
}

```

client.c

```

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

#define BUFSIZE 1024

main()
{
    int clientSock;
    struct sockaddr_in serverAddr;
    socklen_t addr_size;
    char server_ip[] = "127.0.0.1";
    unsigned short server_port=25051;
    char sendBuf[BUFSIZE], recvBuf[BUFSIZE];
    printf("\n\tEnter Sentence:\n\t");
    gets(sendBuf);
    bzero(&serverAddr,sizeof(serverAddr));
    serverAddr.sin_family = AF_INET; //Internet Address family
    serverAddr.sin_port = htons(server_port); //Local Port address
    inet_aton(server_ip,(&serverAddr.sin_addr));
    if((clientSock=socket(AF_INET,SOCK_DGRAM,0))<0)
    {
        printf("\n\tSocket Error.\n");
        exit(1);
    }
    printf("\n\tCLIENT: Socket Created.\n");
    addr_size=sizeof(struct sockaddr_in);
    if(sendto(clientSock,sendBuf,sizeof(sendBuf),0,(struct sockaddr
*)&serverAddr,addr_size)<0)
    {
        printf("\n\tSend Error.\n");
        exit(1);
    }
    printf("\n\tCLIENT: Sent.\n");
    close(clientSock);
}

```

OUTPUT :

Server

```
[student@localhost 5]$ ./server
```

SERVER: Socket Created.

SERVER: Bindind Successfully.

The message received is: My name is Antony Gonsalves.

Number of vowels: 8

Number of consonants: 15

Number of white spaces: 4

Client

[student@localhost 5]\$./client

Enter Sentence:

My name is Antony Gonsalves.

CLIENT: Socket Created.

CLIENT: Sent.