

**Additional Exercise**

|  |  |
| --- | --- |
| Name: | Wilson Vidyut Doloy |
| Reg No: | 19BCE1603 |
| Subject: | CSE1004 Network and Communication Lab |
| Slot: | L52+L653 (Prof. Ganeshan) |
| Date: | 26 August 2020 |

**Error Detection using a Menu driven program:**

Server Side Code:

#include<stdio.h>  
#include<netinet/in.h>  
#include<sys/types.h>  
#include<sys/socket.h>  
#include<netdb.h>  
#include<string.h>  
#include<stdlib.h>  
#define MAX 80  
#define PORT 43454  
#define SA struct sockaddr  
void func(int sockfd)  
{  
int i,length,j,key;  
int clen;  
struct sockaddr\_in cli;  
clen=sizeof(cli);  
for(int j=0;j<1;j++)  
{const char \*ch[1];  
bzero(ch,sizeof(ch));  
recvfrom(sockfd,ch,sizeof(ch),0,(SA \*)&cli,&clen);  
sscanf(ch,"%d", &key);  
if(key==1)  
{char a[MAX],b[MAX];  
char sum[20],complement[20];  
bzero(a,MAX);  
bzero(b,MAX);  
recvfrom(sockfd,a,sizeof(a),0,(SA \*)&cli,&clen);  
recvfrom(sockfd,b,sizeof(b),0,(SA \*)&cli,&clen);  
if(strlen(a)==strlen(b)){  
 length = strlen(a);  
 char carry='0';  
   
 for(i=length-1;i>=0;i--)  
 {  
 if(a[i]=='0' && b[i]=='0' && carry=='0')  
 {  
 sum[i]='0';  
 carry='0';  
 }  
 else if(a[i]=='0' && b[i]=='0' && carry=='1')  
 {  
 sum[i]='1';  
 carry='0';  
   
 }  
 else if(a[i]=='0' && b[i]=='1' && carry=='0')  
 {  
 sum[i]='1';  
 carry='0';  
   
 }  
 else if(a[i]=='0' && b[i]=='1' && carry=='1')  
 {  
 sum[i]='0';  
 carry='1';  
   
 }  
 else if(a[i]=='1' && b[i]=='0' && carry=='0')  
 {  
 sum[i]='1';  
 carry='0';  
   
 }  
 else if(a[i]=='1' && b[i]=='0' && carry=='1')  
 {  
 sum[i]='0';  
 carry='1';  
   
 }  
 else if(a[i]=='1' && b[i]=='1' && carry=='0')  
 {  
 sum[i]='0';  
 carry='1';  
   
 }  
 else if(a[i]=='1' && b[i]=='1' && carry=='1')  
 {  
 sum[i]='1';  
 carry='1';  
   
 }  
 else  
 break;  
 }  
   
 printf("\nSum=%c%s",carry,sum);  
   
 for(i=0;i<length;i++)  
 {  
 if(sum[i]=='0')  
 complement[i]='1';  
 else  
 complement[i]='0'; }  
   
 if(carry=='1')  
 carry='0';  
 else  
 carry='1';  
   
 printf("\nChecksum=%c%s",carry,complement);  
 }  
 else {  
 printf("\nWrong input strings");  
 }  
   
bzero(a,sizeof(a));  
bzero(b,sizeof(b));  
  
  
}  
else if(key==2)  
{  
int keylen,msglen;  
char input[100], key[30],temp[30],quot[100],rem[30],key1[30];  
bzero(input,sizeof(input));  
bzero(key,sizeof(key));  
recvfrom(sockfd,input,sizeof(input),0,(SA \*)&cli,&clen);  
recvfrom(sockfd,key,sizeof(key),0,(SA \*)&cli,&clen);  
keylen=strlen(key);  
msglen=strlen(input);  
strcpy(key1,key);  
for (i=0;i<keylen-1;i++) {  
 input[msglen+i]='0';  
}  
for (i=0;i<keylen;i++)  
 temp[i]=input[i];  
for (i=0;i<msglen;i++) {  
 quot[i]=temp[0];  
 if(quot[i]=='0'){  
 for (j=0;j<keylen;j++){  
 key[j]='0'; }}  
 else{  
 for (j=0;j<keylen;j++){  
 key[j]=key1[j];}}  
 for (j=keylen-1;j>0;j--) {  
 if(temp[j]==key[j])  
 { rem[j-1]='0'; }  
 else{  
 rem[j-1]='1';}  
 }  
 rem[keylen-1]=input[i+keylen];  
 strcpy(temp,rem);  
}  
 strcpy(rem,temp);  
 printf("\nQuotient is ");  
 for (i=0;i<msglen;i++)  
 printf("%c",quot[i]);  
 printf("\nRemainder is ");  
 for (i=0;i<keylen-1;i++)  
 printf("%c",rem[i]);  
 printf("\nFinal data is: ");  
 for (i=0;i<msglen;i++)  
 printf("%c",input[i]);  
 for (i=0;i<keylen-1;i++)  
 printf("%c",rem[i]);  
  
}  
else if(key==3)  
{  
char b[50],pbit;  
int one=0,zero=0;  
bzero(b,sizeof(b));  
recvfrom(sockfd,b,sizeof(b),0,(SA \*)&cli,&clen);  
 pbit=b[strlen(b)-1];  
 for (i = 0; i < strlen(b); i++)  
 {  
 if (b[i] == '1')  
 one++;  
 else  
 zero++;  
 }  
 if(one%2==0 && pbit=='1')  
 printf("The given number is coded in EVEN parity");  
 else if(zero%2==0 && pbit=='0')  
 printf("The given number is coded in ODD parity");  
 else  
 printf("The given number is coded in UNKNOWN parity");  
bzero(b,sizeof(b));  
}  
else{  
printf("choice other than 1, 2 and 3 not accepted");  
}  
bzero(ch,sizeof(ch));  
}  
  
}  
  
int main()  
{  
int sockfd;  
struct sockaddr\_in servaddr;  
sockfd=socket(AF\_INET,SOCK\_DGRAM,0);  
if(sockfd==-1)  
{  
printf("socket creation failed...\n");  
exit(0);  
}  
else  
printf("Socket successfully created..\n");  
bzero(&servaddr,sizeof(servaddr));  
servaddr.sin\_family=AF\_INET;  
servaddr.sin\_addr.s\_addr=htonl(INADDR\_ANY);  
servaddr.sin\_port=htons(PORT);  
if((bind(sockfd,(SA \*)&servaddr,sizeof(servaddr)))!=0)  
{  
printf("socket bind failed...\n");  
exit(0);  
}  
else  
printf("Socket successfully binded..\n");  
func(sockfd);  
close(sockfd);  
}

Client Side Code:

#include<sys/socket.h>  
#include<netdb.h>  
#include<string.h>  
#include<stdlib.h>  
#include<stdio.h>  
#define MAX 80  
#define PORT 43454  
#define SA struct sockaddr  
int main()  
{  
char a[MAX],b[MAX];  
int sockfd,len,n,j,i,key;  
struct sockaddr\_in servaddr;  
sockfd=socket(AF\_INET,SOCK\_DGRAM,0);  
if(sockfd==-1)  
{  
printf("socket creation failed...\n");  
exit(0);  
}  
else{  
printf("Socket successfully created..\n");  
bzero(&servaddr,sizeof(len));  
servaddr.sin\_family=AF\_INET;  
servaddr.sin\_addr.s\_addr=inet\_addr("127.0.0.1");  
servaddr.sin\_port=htons(PORT);  
len=sizeof(servaddr);  
const char \*ch[1];  
for(j=0;j<1;j++)  
{  
printf("\*\*\*\*This is an error detection program\*\*\*\*");  
printf("Enter your choice: ");  
printf("\n1. Checksum\n2. Cyclic Redundancy Check\n3. Parity(1-bit)\n");  
scanf("%c",ch);  
sendto(sockfd,ch,sizeof(ch),0,(SA \*)&servaddr,len);  
sscanf(ch,"%d", &key);  
if(key==1)  
{  
printf("Enter first binary string\n");  
scanf("%s",&a);  
printf("Enter second binary string\n");  
scanf("%s",&b);  
sendto(sockfd,a,sizeof(a),0,(SA \*)&servaddr,len);  
sendto(sockfd,b,sizeof(b),0,(SA \*)&servaddr,len);  
bzero(a,sizeof(a));  
bzero(b,sizeof(b));  
}  
else if(key==2)  
{  
int keylen,msglen;  
char input[100], key[30],temp[30],quot[100],rem[30],key1[30];  
printf("\nEnter Data: ");  
scanf("%s",input);  
printf("\nEnter Key: ");  
scanf("%s",key);  
sendto(sockfd,input,sizeof(input),0,(SA \*)&servaddr,len);  
sendto(sockfd,key,sizeof(key),0,(SA \*)&servaddr,len);  
bzero(input,sizeof(input));  
bzero(key,sizeof(key));  
}  
else if(key==3)  
{  
char b[50];  
printf("\nEnter Coded BCD: ");  
scanf("%s",b);  
sendto(sockfd,b,sizeof(b),0,(SA \*)&servaddr,len);  
bzero(b,sizeof(b));  
  
}  
else{  
printf("choice other than 1, 2 and 3 not accepted");}  
}  
  
  
close(sockfd);  
}  
}

OUTPUT:











