**#CRC**

#include <stdio.h>

#include <string.h>

void xorOperation(char \*dividend, char \*divisor, int start)

{

for (int i = 0; divisor[i] != '\0'; i++)

{

dividend[start + i] = (dividend[start + i] == divisor[i]) ? '0' : '1';

}

}

void computeCRC(char data[], char poly[], char crc[])

{

int dataLen = strlen(data);

int polyLen = strlen(poly);

char temp[200];

strcpy(temp, data);

for (int i = 0; i < polyLen - 1; i++)

{

strcat(temp, "0");

}

for (int i = 0; i <= strlen(temp) - polyLen; i++)

{

if (temp[i] == '1')

{

xorOperation(temp, poly, i);

}

}

strncpy(crc, temp + dataLen, polyLen - 1);

crc[polyLen - 1] = '\0';

}

int main()

{

char data[100], poly[50], crc[50], finalData[200], recvData[200], recvCRC[50];

printf("Enter data to be transmitted:\n");

scanf("%s", data);

printf("\nEnter the Generating polynomial:\n");

scanf("%s", poly);

computeCRC(data, poly, crc);

printf("\nCRC or Check value is : %s\n", crc);

printf("----------------------------------------\n");

strcpy(finalData, data);

strcat(finalData, crc);

printf("Final data to be sent : %s\n", finalData);

printf("----------------------------------------\n");

printf("Enter the received data:\n");

scanf("%s", recvData);

computeCRC(recvData, poly, recvCRC);

printf("\n-----------------------------\n");

printf("Data received: %s\n", recvData);

int error = 0;

for (int i = 0; i < strlen(recvCRC); i++)

{

if (recvCRC[i] != '0')

{

error = 1;

break;

}

}

if (error)

printf("Error detected\n");

else

printf("No error detected\n");

return 0;

}