

1. #include<stdio.h>
2. char m[50],g[50],r[50],q[50],temp[50];
3. void caltrans(int);
4. void crc(int);
5. void calram();
6. void shiftl();
7. int main()
8. {
9. int n,i=0;
10. char ch,flag=0;
11. printf("Enter the frame bits:"); //databits are stored in variable m
12. while((ch=getc(stdin))!='**\n**')
13. m[i++]=ch;
14. n=i; //count of message bits stored in n,eg:3
15. for(i=0;i<16;i++)
16. m[n++]='0'; //after the message bits,remaining 16 zero bits appended
17. m[n]='**\0**';
18. printf("Message after appending 16 zeros:%s",m);
19. for(i=0;i<=16;i++)
20. g[i]='0';
21. g[0]=g[4]=g[11]=g[16]='1';g[17]='**\0**'; //all generator bits set(17 bits)
22. printf("**\n**generator:%s**\n**",g);
23. crc(n); //performs division
24. printf("**\n\n**quotient:%s",q);
25. caltrans(n); //appending the remainder with message bits
26. printf("**\n**transmitted frame:%s",m);
27. printf("**\n**Enter transmitted frame:");
28. scanf("**\n**%s",m);// type the transmitted code
29. printf("CRC checking**\n**");
30. crc(n);
31. printf("**\n\n**last remainder:%s",r);
32. for(i=0;i<16;i++)
33. if(r[i]!='0')
34. flag=1;
35. else
36. continue;
37. if(flag==1)
38. printf("Error during transmission");
39. else
40. printf("**\n\n**Received frame is correct");
41. }
42. void crc(int n)
43. {
44. int i,j;
45. for(i=0;i<n;i++)
46. temp[i]=m[i];
47. for(i=0;i<16;i++)
48. r[i]=m[i];
49. printf("**\n**intermediate remainder**\n**");
50. for(i=0;i<n-16;i++)
51. {
52. if(r[0]=='1') //Only when message bits is 1 we set quotient as 1
53. {
54. q[i]='1';
55. calram();// remainder is calculated
56. }
57. else
58. {
59. q[i]='0';
60. shiftl();
61. }
62. r[16]=m[17+i]; //next bit is brought down from message bits
63. r[17]='**\0**';
64. printf("**\n**remainder %d:%s",i+1,r);
65. for(j=0;j<=17;j++)
66. temp[j]=r[j]; //now the calculated remainder becomes the new temp
67. }
68. q[n-16]='**\0**';
69. }
70. void calram()
71. {
72. int i,j;
73. for(i=1;i<=16;i++)
74. r[i-1]=((int)temp[i]-48)^((int)g[i]-48)+48; //temp starts frm 1 as the 0th position be 1 exor 1=0; remainder starts from 1th position.
75. }
76. void shiftl()
77. {
78. int i;
79. for(i=1;i<=16;i++)
80. r[i-1]=r[i]; //if 0 is the message bit shift to right
81. }
82. void caltrans(int n)
83. {
84. int i,k=0;
85. for(i=n-16;i<n;i++) // 0 bits are substituted with the calculated remainder
86. m[i]=((int)m[i]-48)^((int)r[k++]-48)+48;
87. m[i]='**\0**';
88. }

