# Cyclic Redundancy Check

SOURCE CODE

```
#include<stdio.h>
#include<string.h>
int main(void)
{
   int dat_size, div_size;
   char input[100], quot[100], key[30], rem[30], temp[30], key1[30];
   //input
   printf("Enter the data : ");
   scanf("%s", input);
   printf("Enter the key: ");
   scanf("%s", key);
   dat_size = strlen(input);
   div_size = strlen(key);

   //padding
   for(int i=0; i<div_size-1; i++){
      input[dat_size+i] = '0';
   }
   //temp
   for(int i=0; i<div_size; i++){
      temp[i] = input[i];
   }

   strcpy(key1, key); //copying the key

   //xor
   for(int i=0; i<dat_size; i++){
      quot[i] = temp[0];
      if(quot[i]=='0'){
```

```
            for(int j=0; j<div_size; j++)
                key[j] = '0';
        }
        else{
            for(int j=0; j<div_size; j++)
                key[j] = key1[j];
        }
        for(int j=div_size-1; j>0; j--){
            if(key[j]==temp[j])
                rem[j-1] = '0';
            else
                rem[j-1] = '1';
        }
        rem[div_size-1] = input[i+div_size];
        strcpy(temp, rem);
    }
    strcpy(rem, temp);

    printf("\nThe remainder is: ");
    for(int i=0; i<div_size; i++)
        printf("%c", rem[i]);
    printf("\nNew Data is: ");
    for(int i=0; i<dat_size; i++)
        printf("%c", input[i]);
    for(int i=0; i<div_size; i++)
        printf("%c", rem[i]);

    return 0;
}
```

OUTPUT (on white page)