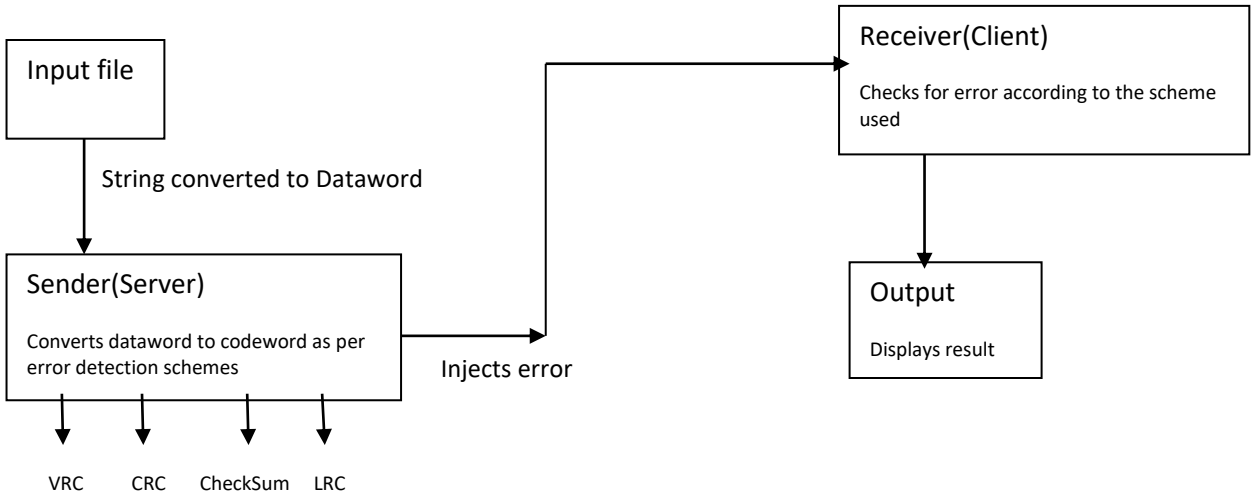
Aryapriyo Mandal
Roll=001910501033(A2)

3rd Year 1st Sem BCSE UG
Computer Network
Assignment 1

Problem Statement:-Design and implement an error detection module.

Description:- Design and implement an error detection module which has four schemes namely LRC, VRC, Checksum and CRC. The Sender program should accept the name of a test file (contains binary strings) from the command line. Then it will prepare the data frame from the input. Based on the schemes, codeword will be prepared. Sender will send the codeword to the Receiver. Receiver will extract the dataword from codeword and show if there is any error detected.

Design:- Network Communication is established and simulated using Socket programming through client-server. Code is written in Java. The data from a text file is converted to a 32 bit binary string(with padding) to obtain the dataword and is then stored in another text file. The Sender program(Server) takes the string from the file, encodes it with the detection scheme selected by the user, injects error randomly of length decided by the user and then writes it to file again which is sent to the Receiver(Client). The Receiver(Client) checks for the error using the same scheme and prints whether there is error or not. Exhanges of data occur using socket.



Input/Output : 32 bit binary strings are used as data. All the schemes (LRC,VRC,CRC,Checksum) generates respective codewords. Client program returns true/false bsed on error is detected or not by those corresponding scheme.

Assumption made is that the dataword generated is not more than 32 bit.

Implementation:-

Server program

Classes and Functions used:- Sender.java(class)

Functions used are:

- read_file()- Reads the text from file and stores it in a string.
- get_dataword(String str)- Converts a string to binary equivalent and returns a string of its binary values padded to 32 bits.
- write_to_file(String str)- Writes a string message to a file which is sent to the receiver.
- send(String str)- Sends the file containing error injected codeword to the receiver. It implements socket programming using the serversocket class. It basically writes a string to a filewhich is essentially the codeword plus error which is transmitted to the receiver.
- vrc_code(int arr[])- Returns the vrc parity bit of the given dataword.
- crc_code(int arr[])- Returns the crc code of the given dataword.
- Divide(int div[],int divisor[],int rem[])- Used for crc calculation.
- int[] lrcarray(int[] seq,int k)- Returns the lrc code for a given dataword.
- int[] checksum(int[] seq, int k)- Returns the checksum code of a given dataword.
- changebit(String str, int pos)- Injects error in the index pos.
- get_error(String str)- Takes the length of burst errors from the users and then injects error in it.

main()- The main method creates an object of the Sender class and converts the given text from a file to the dataword. It then takes entries from the user asking which coding scheme to use. The codeword is prepared based on the coding scheme and then random error from the user of their choice burst length is injected into the codeword and then it is sent to the receiver.

Client program

Classes and Functions used:- Receiver.java(class)

Functions used are:

- read_file()- Reads the text from file and stores it in a string.

- truelength(String str)- Reads the actual length of a string from a file
- receive()- Receives the error injected codeword from the sender and wirtes it down in a file. It uses the Client Socket to establish the communication between the sender and receiver class.
- check_vrc(int arr[])- Checks if there was error in the codeword transmitted and prints the erro message accordingly.
- check_crc(int arr[])- Checks if there is any error in the crc codeword and displays an erro r message accordingly.
- int[] lrcarray(int[] seq,int k)- Returns the lrc code for a given dataword.
- check_lrc(int arr[], int k, int lrc[])- checks if the lrccodeword is erroneous or not.
- isValidChecksum(int arr[], int k, int cksum[])- checks if the checksumcodeword is erroneous or not.

main()- It reads the received file and then checks for the error according the necessary coding schemes.

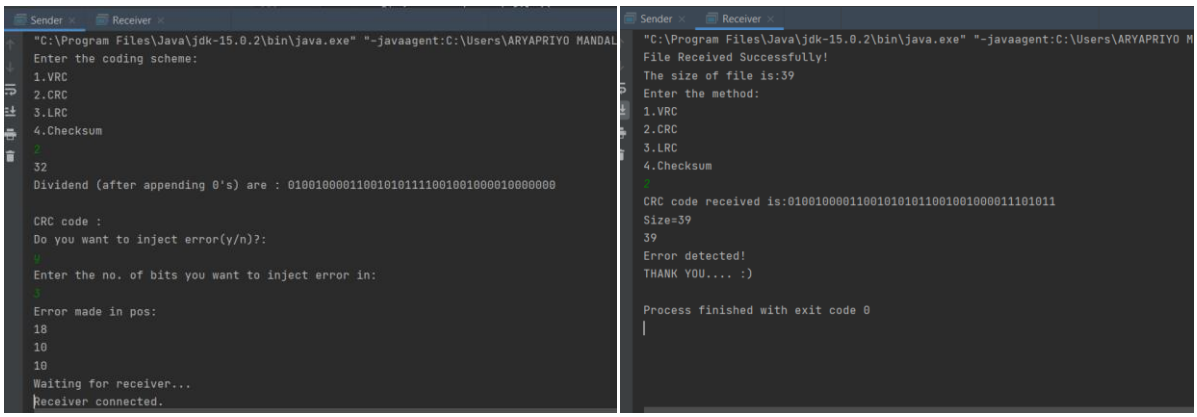
Note: The length of the dataword cannot be be more than 32 bits asper the code rather for this particular implementation of the code. This is one drawback of the design. Another drawback is that it does not check repeatedly as the length of the dataword is too small. And therefore a sharp and detailed comparative study wasn't possible.

Test Cases:-

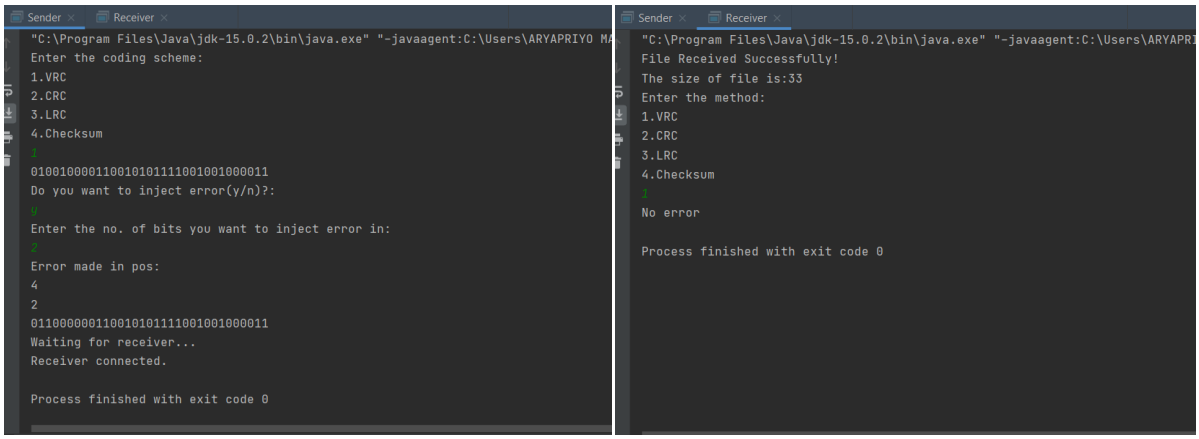
5 Test Cases for each error detecting scheme has been manually checked. Error is injected in the Sender side in the codeword portion.

Some Output test cases are shown as follows:-

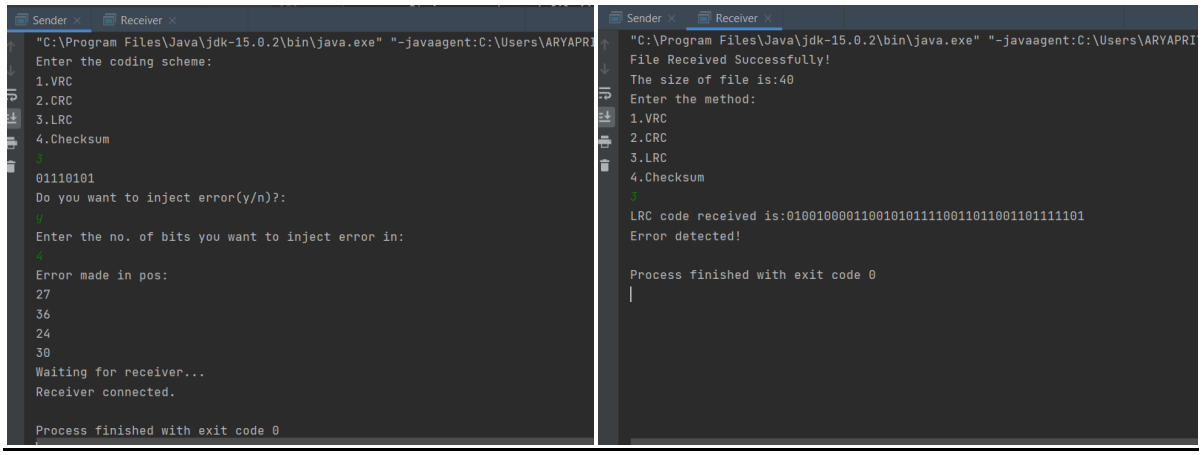
CRC:



VRC:

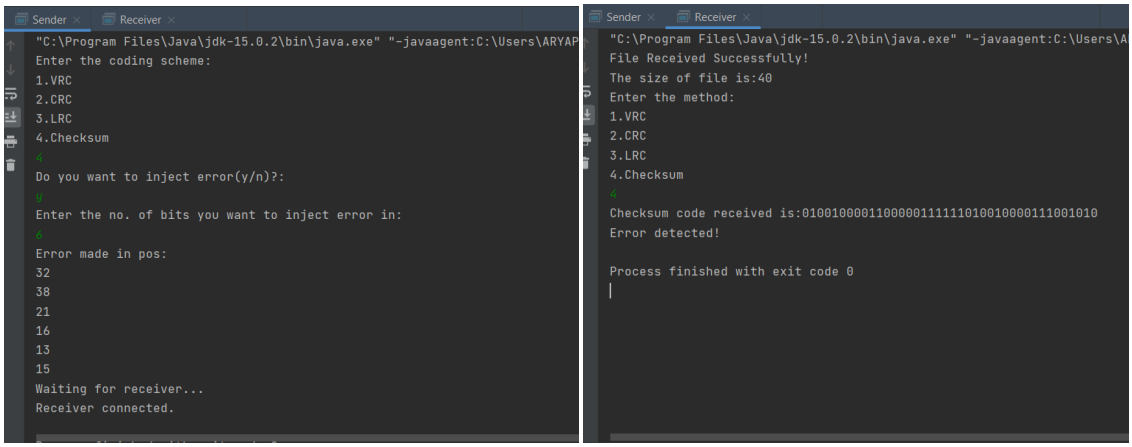


LRC:



(for error in positions 0 and 8 the lrc wasn't able to predict the right error)

CheckSum:



(for error in position 2 and 10 checksum 2 wasn't able to detect he right error).

Coding scheme	No. of errors detected out of 5 test cases
VRC	3
CRC	5
LRC	4
CheckSum	4

Result and Analysis:-

- VRC can detect all single bit errors correctly. It can also detect burst errors of length odd. Its accuracy in real life scenario is 52.61%.
- CRC can detect all single bit errors and most burst errors. In real life it has an accuracy of 99.99%.
- LRC can detect all single bit errors. It can also detect most burst errors. But it fails to detect errors if two bits in a data unit are damaged and two bits in exactly the same position in other data unit are also damaged. In real life it has an accuracy of 99.73%
- CheckSum can detect all single bit errors. It can also detect most burst errors. It fails to detect errors if the value of one data item is increased (intentionally or maliciously) and the value of another one is decreased (intentionally or maliciously) the same amount. In real life its accuracy is 99.84%.

Conclusion:-

Among all the schemes used, CRC is by far the most sensitive to errors and therefore the best among the four, followed by checksum, lrc and vrc. However greater accuracy can be achieved by using all the four schemes together.