CHAPTER 7
SOURCECODE

```
import java.util.Arrays;
importjava.util.Scanner;

publicclassBitwiseOperations{

   //Convertinputtoabinaryarray
   publicstaticint[]convertToBinaryArray(Stringinput,intn){
      //Trytointerpretthe input asabinarystring try {
         //Ensureit'savalidbinarystringandthenconvertto anarrayofintegers if
         (input.matches("[01]+")) {
            int[]binaryArray=newint[Math.min(input.length(),n)]; for
            (int i = 0; i < binaryArray.length; i++) {
               binaryArray[i]=Character.getNumericValue(input.charAt(i));
            }
            returnbinaryArray;
         }
      }catch(Exceptione){
         //Continuetothenextmethodifthisfails
      }

      //Trytointerpretthe input asastring(convertto binaryASCII) try {
         byte[]bytes= input.getBytes();
         StringBuilderbinaryString=newStringBuilder(); for
         (byte b : bytes) {
            StringbinaryChar=String.format("%8s",Integer.toBinaryString(b&0xFF)).replace(",

'0');          binaryString.append(binaryChar);
         }
         returnconvertToBinaryArray(binaryString.toString(),n);
      }catch(Exceptione){
         //Continuetothenextmethodifthisfails
      }

      //Trytointerprettheinput asanumber try {
         intnum= Integer.parseInt(input);
         StringbinaryString=Integer.toBinaryString(num); return
         convertToBinaryArray(binaryString, n);
      }catch(Exceptione){
         //Throw anerrorifnone ofthe abovemethods work
         thrownewIllegalArgumentException("Invalidinput:Must beabinarystring,aregularstring, or a
number.");
      }
   }
```

```
publicstaticint[]bitwiseOperations(int[]BMP1,int[]BMP2,intn,int W1){
   //Ensurebotharraysareoflengthn
   BMP1=Arrays.copyOf(BMP1,n);
   BMP2=Arrays.copyOf(BMP2,n);

   //PerformtheANDoperation int[]
   R = new int[n];
   for(inti=0;i<n; i++){
      R[i]=BMP1[i] &BMP2[i];
   }

   //UpdateBMP1andBMP2usingXORwithR for
   (int i = 0; i < n; i++) {
      BMP1[i]=BMP1[i]^R[i];
      BMP2[i]=BMP2[i]^R[i];
   }

   //CalculatethesumofRandcompare itto W1 int
   sumR = 0;
   for(intvalue:R){
      sumR += value;
   }

   returnsumR>W1?R:newint[0];
}

public static void main(String[] args) {
   Scannerscanner=newScanner(System.in);

   // Input size of
   BMP1System.out.println("EnterthesizeofBMP
   1:"); int sizeBMP1 = scanner.nextInt();

   //Input BMP1
   System.out.println("EnterBMP1(binarystring,regularstring,ornumber):"); String
   bmp1Input = scanner.next();

   // Input size of
   BMP2System.out.println("EnterthesizeofBMP
   2:"); int sizeBMP2 = scanner.nextInt();

   //Input BMP2
   System.out.println("EnterBMP2(binarystring,regularstring,ornumber):"); String
   bmp2Input = scanner.next();

   //Input n
   System.out.println("Enterthenumberofbitstoprocess(n):"); int n
   = scanner.nextInt();

   //Input W1
   System.out.println("Enterthethresholdvalue(W1):");
```

```
        intW1=scanner.nextInt();

        //Convertinputstobinaryarrays
        int[]BMP1=convertToBinaryArray(bmp1Input,sizeBMP1);
        int[]BMP2=convertToBinaryArray(bmp2Input,sizeBMP2);

        //Runthebitwiseoperations try
        {
            int[] result = bitwiseOperations(BMP1, BMP2, n, W1);
            System.out.println("IB_Results:"+Arrays.toString(result));
        } catch (IllegalArgumentException e) {
            System.out.println("Error:"+e.getMessage());
        }

        scanner.close();
    }
}
```