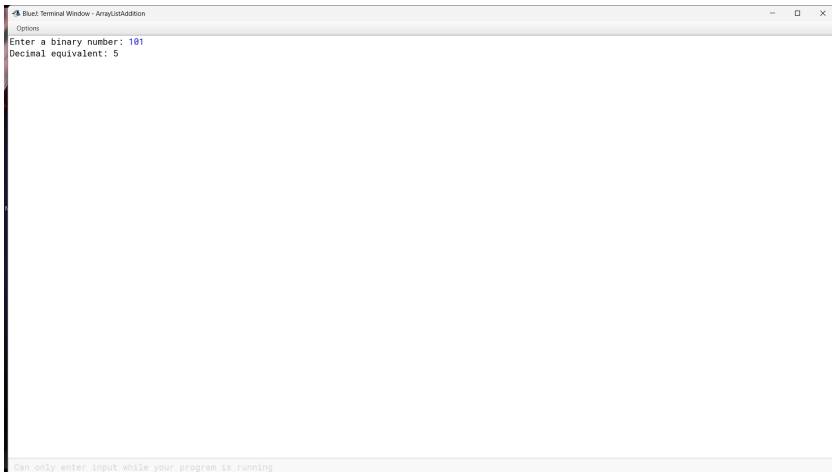
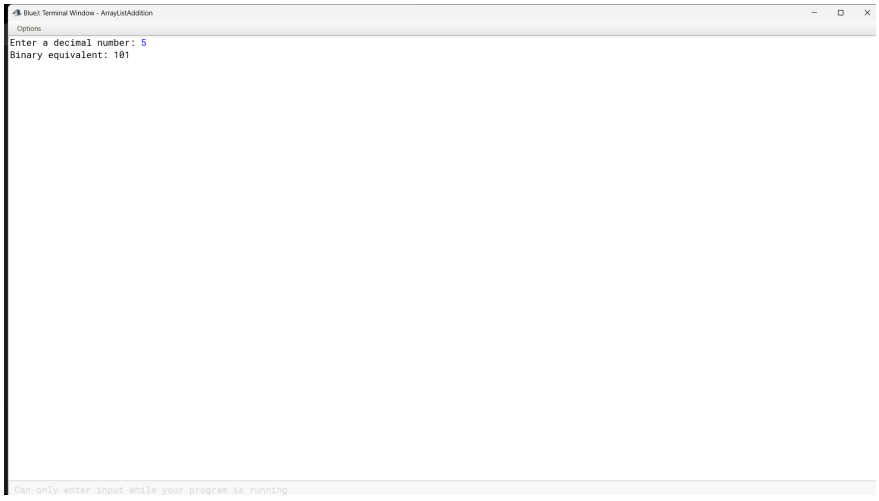
1)

```
import java.util.Scanner;

public class BinaryToDecimal {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);


        System.out.print("Enter a binary number: ");
        String binaryStr = input.nextLine();


        int decimal = 0;
        int base = 1;
        for (int i = binaryStr.length() - 1; i >= 0; i--) {
            if (binaryStr.charAt(i) == '1') {
                decimal += base;
            }
            base *= 2;
        }


        System.out.println("Decimal equivalent: " + decimal);
    }
}
```



2)import java.util.Scanner;

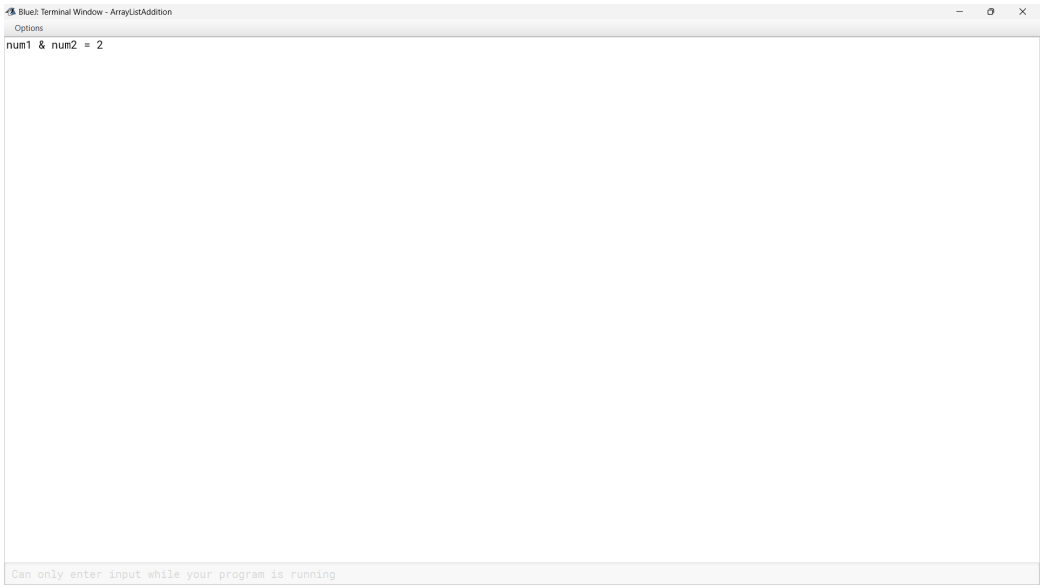
```
public class DecimalToBinary {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        // Read the decimal number from the user
        System.out.print("Enter a decimal number: ");
        int decimal = input.nextInt();

        // Convert the decimal number to binary
        StringBuilder binary = new StringBuilder();
        while (decimal > 0) {
            int remainder = decimal % 2;
            binary.append(remainder);
            decimal /= 2;
        }

        // Reverse the binary string
        binary.reverse();

        // Print the binary equivalent
        System.out.println("Binary equivalent: " + binary);
    }
}
```



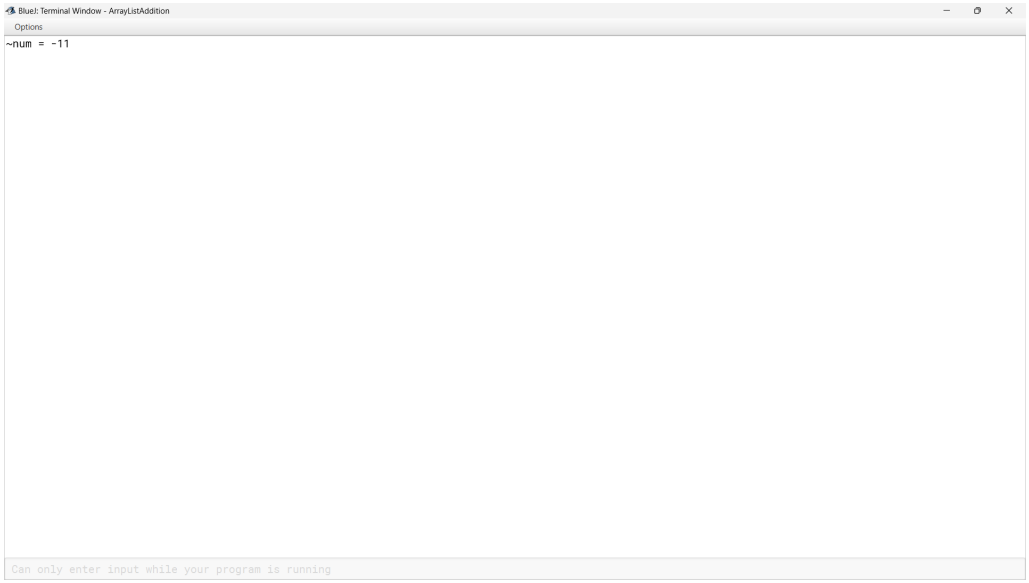
3)public class BitwiseAndExample {

```
    public static void main(String[] args) {
        int num1 = 10;
```

```
        int num2 = 6;

        int result = num1 & num2;

        System.out.println("num1 & num2 = " + result);
    }
}
```

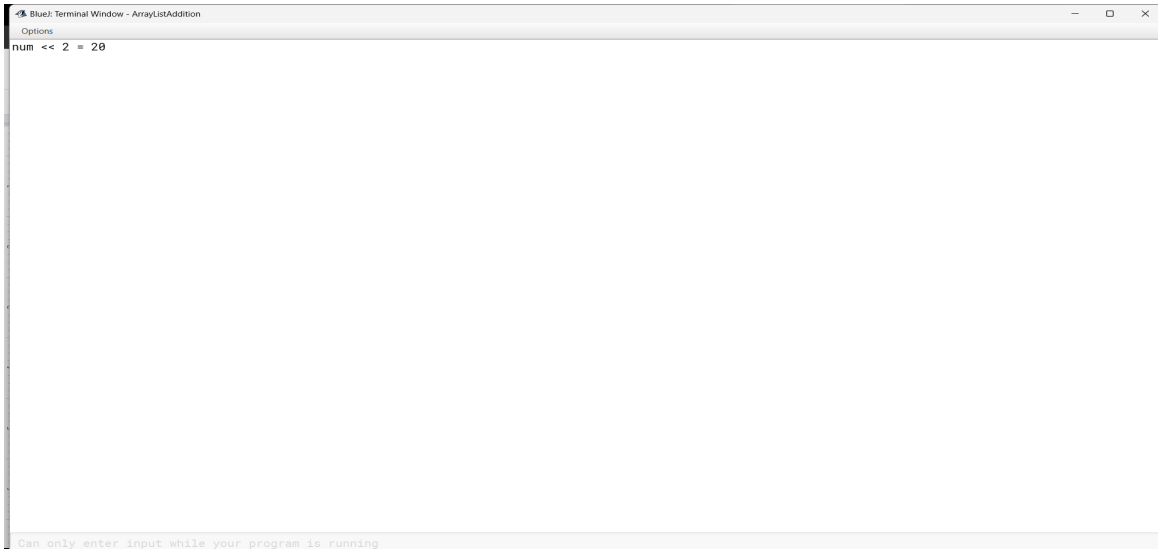



```
4)public class BitwiseComplementExample {
    public static void main(String[] args) {
        int num = 10; // binary representation: 0000 1010

        int complement = ~num;

        System.out.println("~num = " + complement);
    }
}
```



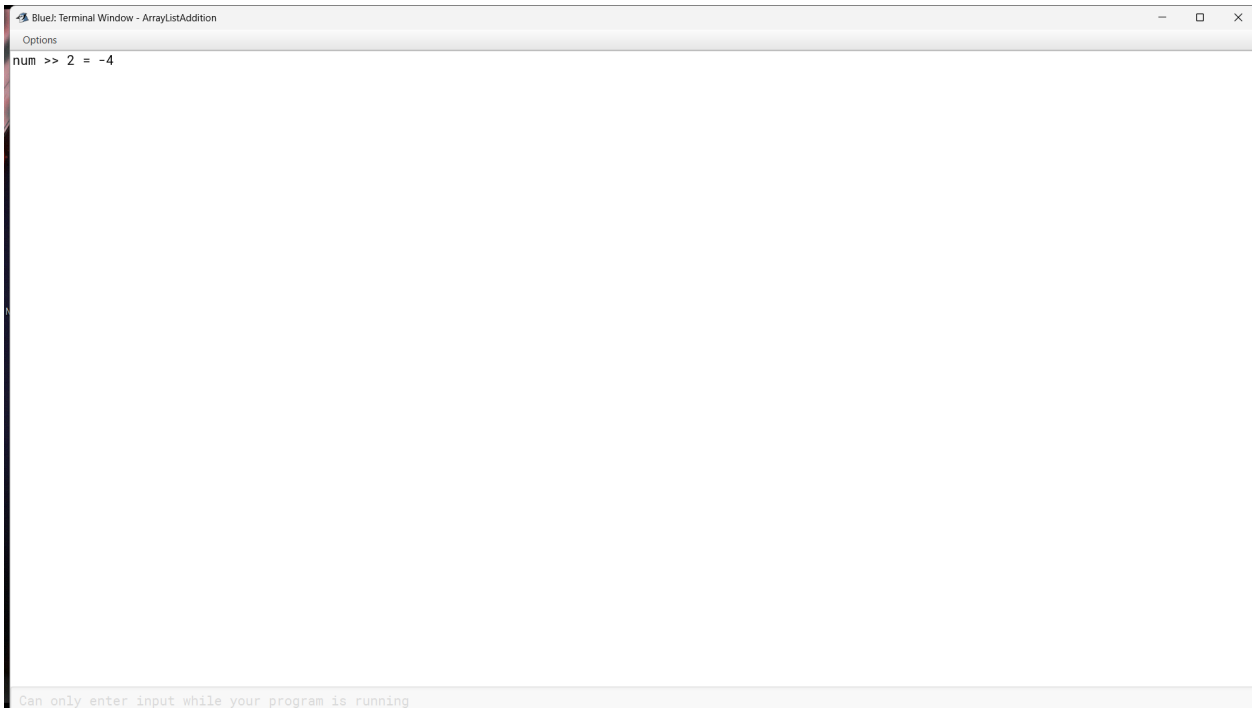
5)

```
public class LeftShiftExample {
    public static void main(String[] args) {
        int num = 5;

        int result = num << 2;

        System.out.println("num << 2 = " + result);
    }
}
```



6)

```
public class SignedRightShiftExample {
    public static void main(String[] args) {
        int num = -15; // binary representation: 1111 0001

        int result = num >> 2;

        System.out.println("num >> 2 = " + result); // prints -4 (binary representation: 1111 1100)
    }
}
```



Debugging

1)

```
interface Square {
  int calculate(int x);
}

class Test {
  public static void main(String args[]) {
     Square s = (int x) -> x * x;

    int a = 5;
    int ans = s.calculate(a);
    System.out.println(ans);
  }}
```

2)
```
class GFG {
  public static void main(String[] args) {
    int[] arr = new int[5];
    arr[0] = 10;
    arr[1] = 20;
    arr[2] = 30;
    arr[3] = 40;
    arr[4] = 50;
      for (int i = 0; i < arr.length; i++) {
      System.out.println("Element at index " + i + ": " + arr[i]);
    }
  }
}
```

3)
```
class parent {
    void show() {}
}

class child extends parent {
    void show() {}
}
```