

1)

```
import java.util.Scanner;

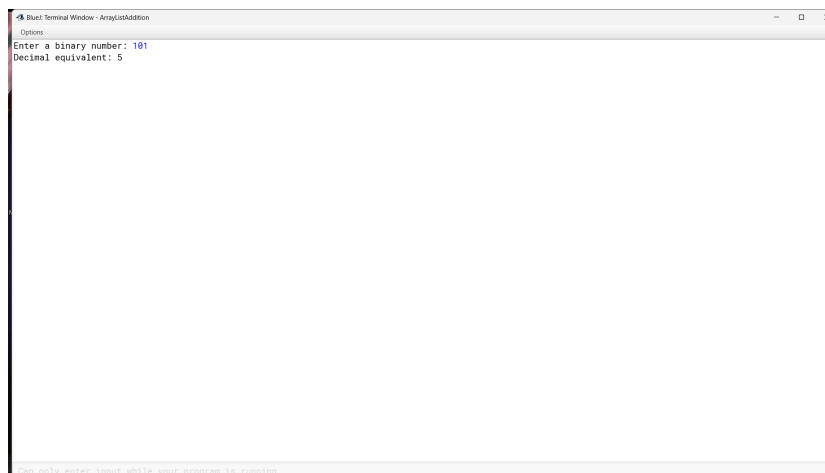
public class BinaryToDecimal {

    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);
    }
}
```

A screenshot of a Java IDE terminal window. The window title is "BlueJ Terminal Window - ArrayAddition". It shows the execution of the program. The prompt "Enter a binary number: " is followed by the input "101". The output is "Decimal equivalent: 5". At the bottom, a message says "Can only enter input while your program is running".

```
BlueJ Terminal Window - ArrayAddition
Options
Enter a binary number: 101
Decimal equivalent: 5

Can only enter input while your program is running
```

```

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);
    }
}

```

The screenshot shows a terminal window titled "Blue Terminal Window - AnyListAddition". The prompt "Options:" is visible. The user has entered "5" in response to the prompt "Enter a decimal number:". The program has executed and displayed the output "Binary equivalent: 101". At the bottom of the terminal, a message states "Can only enter input while your program is running".

```

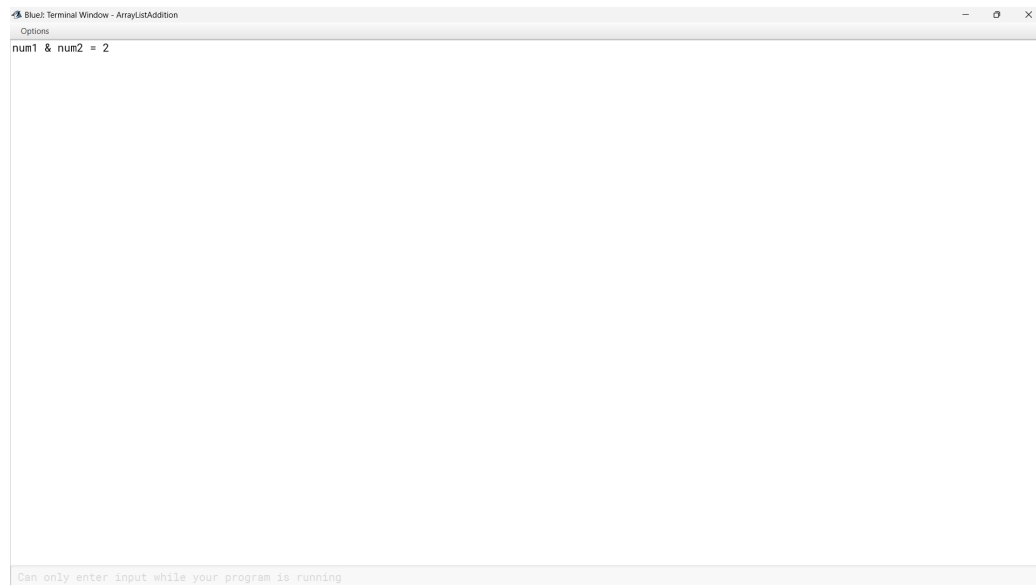
3)public class BitwiseAndExample {
    public static void main(String[] args) {
        int num1 = 10;

```

```
int num2 = 6;

int result = num1 & num2;

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

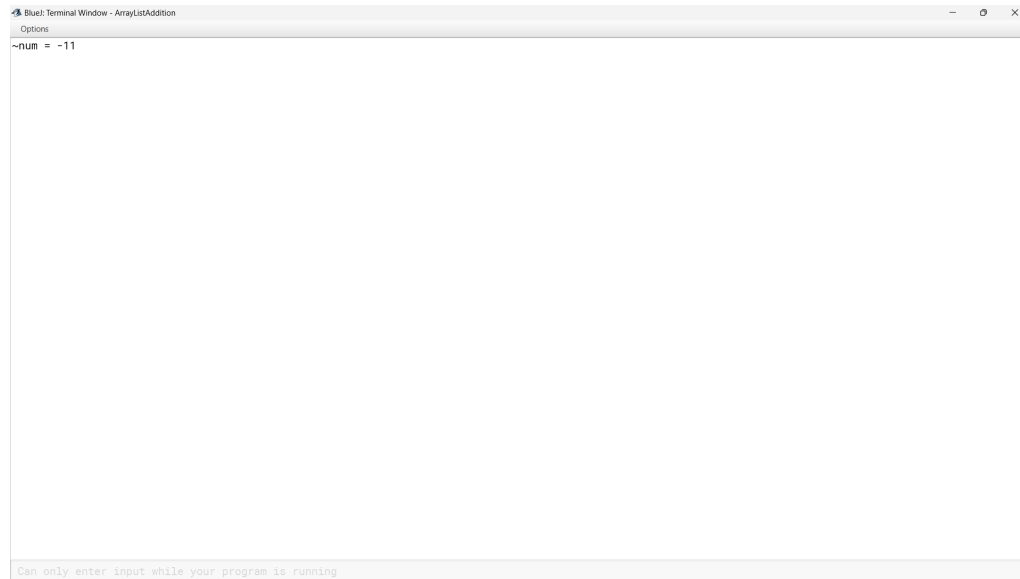
A screenshot of a BlueJ Terminal Window titled "BlueJ Terminal Window - ArrayListAddition". The window has a menu bar with "Options" and standard window controls (minimize, maximize, close). The main area displays the output "num1 & num2 = 2". At the bottom, a status bar reads "Can only enter input while your program is running".

```
BlueJ Terminal Window - ArrayListAddition
Options
num1 & num2 = 2
Can only enter input while your program is running
```

```
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);
    }
}
```

A terminal window titled "BlueJ: Terminal Window - ArrayListAddition" with an "Options" menu. The main area displays the text "~num = -11". At the bottom, a status bar reads "Can only enter input while your program is running".

```
BlueJ: Terminal Window - ArrayListAddition
Options
~num = -11

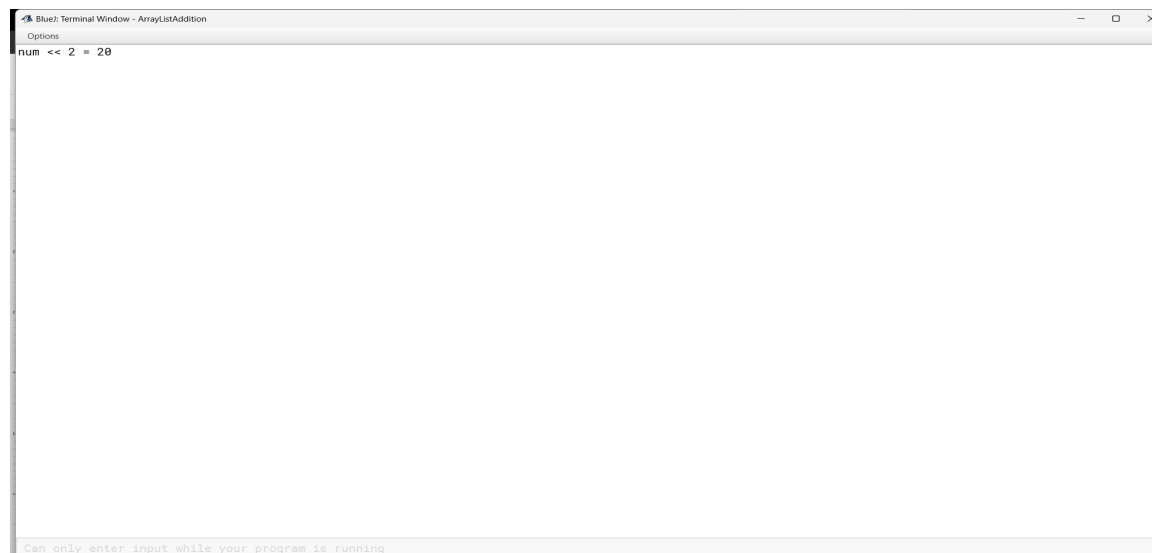
Can only enter input while your program is running
```

5)

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

        int result = num << 2;

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

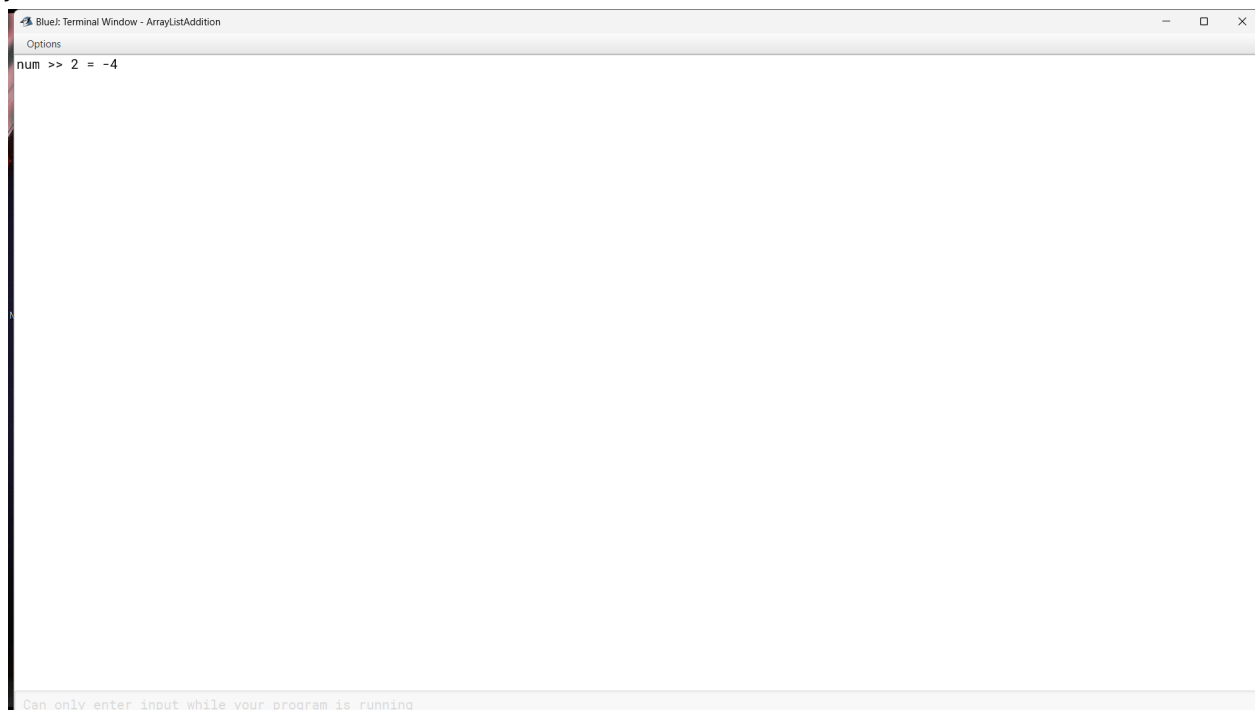
A terminal window titled "BlueJ: Terminal Window - ArrayListAddition" with an "Options" menu. The main area displays the text "num << 2 = 20". At the bottom, a status bar reads "Can only enter input while your program is running".

```
BlueJ: Terminal Window - ArrayListAddition
Options
num << 2 = 20

Can only enter input while your program is running
```

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() {}  
}
```