```
public class Fibonacci
{
    public static void main(String[] args)
    {

        //number of elements to generate in the sequence
                int max = 15;

                // create the array to hold the sequence of Fibonacci
numbers
                int[] sequence = new int[max];

                //create the first 2 Fibonacci sequence elements
                sequence[0] = 0;
                sequence[1] = 1;

                //create the Fibonacci sequence and store it in int[]
sequence
                for(int i = 2; i < max; i++)
                {
                    sequence[i] = sequence[i - 1] + sequence[i - 2];
                }


                //print the Fibonacci sequence numbers
                System.out.println("The first " + max + " elements in
the Fibonacci sequence are: ");
        for(int i = 0; i < max; i++)
        {
            System.out.print(sequence[i] + " ");
        }

        System.out.println("\nThe element after 21 is " +
findNextElement(sequence, 21));
        System.out.println("The element after 233 is " +
findNextElement(sequence, 233));

        // Be careful! Where is 377 in the array? How should your
function avoid problems?
        System.out.println("The element after 377 is " +
findNextElement(sequence, 377));



    }

    // This method returns the element that comes after element
'toFind'
    public static int findNextElement (int[] arr, int toFind)
    {
```

```
        if(arr == null)
        {
            return -1;
        }

        int i = 0;

        while(i < arr.length - 1)
        {
            if(arr[i] == toFind){
                return arr[i + 1];
            }

            i++;
        }


        return -1;

    }
}
```