### Q. Java Program to find Armstrong numbers with Example

How to check if a number is Armstrong number or not? or write a Java program to find Armstrong's number? This is a common Java interview question asked on-campus interviews and fresher level interviews. This is also a popular Java programming exercise on various schools, colleges and computer courses to build programming logic among Students. An Armstrong number is a 3 digit number for which the sum of cube of its digits is equal to the number itself. One of the popular examples of the Armstrong number is 153 as 153= 1+ 125+27 which is equal to 1^3+5^3+3^3 which is equal to the sum of cube of its digits 1, 5, and 3  
One more example of the Armstrong number is 371 because it is the sum of 27 + 343 + 1 which is equal to 3^3 + 7^3 + 1^3. In this Java program example, we will see a complete code example of Java program to check if any 3 digit number is Armstrong number or not.  
If you are going for Java interview, then be prepared for some follow-up questions e.g. finding prime numbers, or finding Armstrong number of more than 3 digits.  
Btw, good knowledge of essential programming concepts, operators, algorithms, data structure, and logic is important for any software engineer, particularly to do well on technical interviews.  If you are preparing for a tech interview then you should double down on these topics.

If you need recommendations to refresh your Data Structure and algorithm skills then I highly recommend checking out Data Structures and Algorithms: Deep Dive Using Java course on Udemy. It's a hands-on course and covers all essential data structures. It's also very affordable and you can get in just $10 on Udemy flash sales which happen every now and then.

Here is complete code for checking if a number is Armstrong number or not. It uses a method called isArmstrong(int number) to implement the logic for checking if a number is Armstrong nor not.  
  
Btw, this is not the same program as print all Armstrong numbers between 0 and 999 but you can use this logic to solve that question as well. All you need to do is loop till 1000 and check if the number is Armstrong or not. If yes then print otherwise move to next number.  
  
By the way, this Java program is in continuation of our earlier programming exercise like

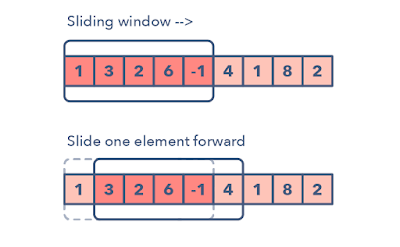
Java program to reverse String in Java,

Java program to reverse a number in Java and

Java program to print the Fibonacci number in Java.

Program for calculating the factorial of a number using recursion in Java.

These programs will help you to build the coding sense required to solve coding problems on interviews and your day to day life. It's also one of the important skills to map a new coding problem into already known one and essential to pass the coding interview on companies like Google, Facebook, Amazon, and Microsoft.  
  
If you are preparing for those companies then I also suggest you go through Grokking the Coding Interview: Patterns for Coding Questions on Educative, an interactive portal for coding interviews to learn some 16 useful coding patterns like Sliding Window, Two Pointers, Fast and Slow Pointers, Merge Intervals, Cyclic Sort, and Top K elements that can help you to solve zones of frequently asked coding problems.

[](https://www.educative.io/collection/5668639101419520/5671464854355968?affiliate_id=5073518643380224)

Java Program to Find Armstrong Number

import java.util.Scanner;

/\*\*

\* Simple Java Program to check or find if a number is Armstrong number or not.

\* An Armstrong number of three digit is a number whose sum of cubes of its digit is equal

\* to its number. For example 153 is an Armstrong number of 3 digit because 1^3+5^3+3^3 or 1+125+27=153

\* @author Javin

\*/

public class ArmstrongTest{

public static void main(String args[]) {

//input number to check if its Armstrong number

System.out.println("Please enter a 3 digit number to find if

its an Armstrong number:");

int number = new Scanner(System.in).nextInt();

//printing result

if(isArmStrong(number)){

System.out.println("Number : " + number + " is an Armstrong number");

}else{

System.out.println("Number : " + number + " is not an Armstrong number");

}

}

/\*

\* @return true if number is Armstrong number or return false

\*/

private static boolean isArmStrong(int number) {

int result = 0;

int orig = number;

while(number != 0){

int remainder = number%10;

result = result + remainder\*remainder\*remainder;

number = number/10;

}

//number is Armstrong return true

if(orig == result){

return true;

}

return false;

}

}

Output:

Please enter a 3 digit number to find if its an Armstrong number:

153

Number : 153 is an Armstrong number

Please enter a 3 digit number to find if its an Armstrong number:

153

Number : 153 is an Armstrong number

Please enter a 3 digit number to find if its an Armstrong number:

371

Number : 371 is an Armstrong number

That's all on How to check if a number is Armstrong in Java. It’s a pretty simple Java program and if you look closely it just gets digit by digit by using the remainder operator and reduce the number by 1 digit after dividing it by 10. Let me know if you find any bug on this Java program to check for the Armstrong number.

How to Check if given Array contains a value in Java - Linear Search vs Binary Search

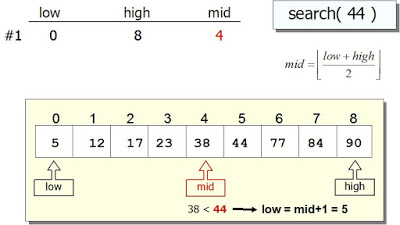
Hello guys, one of the common coding questions from Java interviews is how to test if an Array contains a certain value or not? This is a simple question, but sometimes interview pressure makes candidates nervous. In this article, you'll learn how to solve this problem using different approaches. Since an array in Java doesn't have any inbuilt method for searching values, interviewers prefer to ask this question, to see how a candidate deals with such a situation. If you have good knowledge of Java API, then you will immediately come to know that there are alternatives available like using the binarySearch() method of Java Java .util.Arrays class or taking advantage of ArrayList contains method by first converting your array to ArrayList.  
  
If you come up with those solutions, the Interviewer will surely ask you to write down a method to search an element in an array without using any library method. For that, you need to remember your Data structure and algorithm classes. You can easily solve this question if you know the linear search or binary search algorithm.  
  
Linear search is very simple to implement, all you need to do is loop through the array and check each value if that is the one or not.  
  
Binary search is a little tricky but not too difficult either, the recursive solution is very natural as well. In this tutorial, though I have given two solutions, one is using ArrayList, and the second is using linear search, leaving binary search an exercise for you. But you must remember to sort the array before using binary search.  
  
That's where a good knowledge of essential Data Structure and Algorithms comes to rescue. I strongly suggest every Java developer brush up their Data structure and algorithm skills before going for any kind of programming interview. If you need a resource, I recommend Data Structures and Algorithms: Deep Dive Using Java course by Tim Buchalaka on Udemy.  
  
It's very affordable and covers all essential data structures like the array, linked list, binary tree, hash table, stack, queue, and others.

How to check if an array contains a value in Java?

To give you some more idea of the problem, let's see an example first; suppose you have a String[] with values like so:

public static final String[] names = new String[] {"Java","JEE","Scala","C++"};

Given String name, you need to return true or false, depending upon whether names contain that value or not. By the way, here is a full example of how to search a number on integer array and searching for a name on String array, in case you need any help.  
  
This example contains two methods isExists() and contains(), which returns true if the value is present in the array. The first method uses contains() method of ArrayList by first converting given an array to ArrayList, while the second method simply uses a linear search algorithm to search on a Java array.  
  
By the way, to make the question more challenging, I usually asked the candidate to write a parametric method using generic so that it will work for any type of object array in Java. If you are interested, you can try solving that version as well.  
  
In case, if you are not familiar with Generics and writing Parametric class and methods in Java then I suggest you join a comprehensive Java course like The Complete Java MasterClass on Udemy to learn Generics better. It's very important for writing production-quality Java code.

[](https://1.bp.blogspot.com/-xhoUcvGGWNg/XpZ1qczYQzI/AAAAAAAAeJw/MQN244LO6qQDQ-XCoMWn4gn1jlKyzlmlwCLcBGAsYHQ/s1600/Finding%2BMissing%2BNumber%2Bin%2BJava%2Busing%2BBinary%2BSearch%2BAlgorithm.jpg)

Java Program to Search String or Integer in Given Array

Without wasting any more of your time, here is our complete Java program to search any given String or integer value in the given array. If you are using Eclipse IDE, just copy-paste the code and run it, you don't need to create Java source file, Eclipse will take care of that, provided you have selected a Java project.

import java.util.Arrays;

/\*\*

\* Java Program to check if an array contains a value or not. Basically this program tells you

\* how to search for an element in array, it could be an integer number or String value.

\*

\* @author Javin Paul

\*/

public class ArrayTest{

public static void main(String args[]) {

//test our method to see if array contains a certain value or not

Integer[] input = new Integer[]{1, 2, 3, 4, 5};

System.out.printf("Does array %s has %s? %b %n",

Arrays.toString(input), 5, isExists(input, 5));

System.out.printf("Does array %s contains %s? %b %n",

Arrays.toString(input), 5, contains(input, 5));

System.out.printf("Does array %s has %s? %b %n",

Arrays.toString(input), 6, isExists(input, 6));

System.out.printf("Does Integer array %s contains %s? %b %n",

Arrays.toString(input), 6, contains(input, 6));

String[] names = new String[]{"JP", "KP", "RP", "OP", "SP"};

System.out.printf("Does array %s has %s? %b %n",

Arrays.toString(names), "JP", isExists(names, "JP"));

System.out.printf("Does String array %s contains %s? %b %n",

Arrays.toString(names), "JP", contains(names, "JP"));

System.out.printf("Does array of names %s has %s? %b %n",

Arrays.toString(names), "MP", isExists(names, "MP"));

System.out.printf("Does array %s contains %s? %b %n",

Arrays.toString(names), "UP", contains(names, "UP"));

}

/\*\*

\* Function to test if Array contains a certain value or not.

\* This method take advantage of

\* contains() method of ArrayList class, by converting array to ArrayList.

\*

\* @return true if array contains

\*/

public static <T> boolean isExists(final T[] array, final T object) {

return Arrays.asList(array).contains(object);

}

/\*\*

\* Another method to search an item in Java array.

\* This method loop through array and use

\* equals() method to search element. This actually performs

\* a linear search over array in Java

\*

\*@return true if array has provided value.

\*/

public static <T> boolean contains(final T[] array, final T v) {

for (final T e : array) {

if (e == v || v != null && v.equals(e)) {

return true;

}

}

return false;

}

}

Output:

Does array [1, 2, 3, 4, 5] has 5? true

Does array [1, 2, 3, 4, 5] contains 5? true

Does array [1, 2, 3, 4, 5] has 6? false

Does Integer array [1, 2, 3, 4, 5] contains 6? false

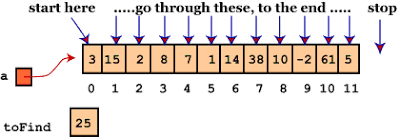
Does array [JP, KP, RP, OP, SP] has JP? true

Does String array [JP, KP, RP, OP, SP] contains JP? true

Does array of names [JP, KP, RP, OP, SP] has MP? false

Does array [JP, KP, RP, OP, SP] contains UP? false

You can see the result as true or false if the array contains a particular value or not. Like in the first output array contains 5 so the result is true, but in the third example, the array doesn't contain 6, so the result is false.

[](https://4.bp.blogspot.com/-Ig8byh1krmI/VHHjLqMoF-I/AAAAAAAACHA/PSkMZdXNMPQ/s1600/linear-search-array.gif)

That's all on how to find if an array contains a particular value or not. As I told you, if you are allowed to use Java API, then you can either use the binarySearch() method of Java Java .util.Arrays class or you can simply convert your array to ArrayList and then call its contains() method.  
  
If using Java API or any third party is not allowed, then you can write your own function to search an element in an array using the either binary search or linear search method. If you write binary search, then be prepared with both iterative and recursive methods, as the Interviewer will more likely to ask both of them.

How to Find Duplicate Characters on String - Java Programming Problems

Hello guys, today's programming exercise is to write a program to find repeated characters in a String. For example, if given input to your program is "Java", it should print all duplicates characters, i.e. characters appear more than once in String and their count like a = 2 because of character 'a' has appeared twice in String "Java". This is also a very popular coding question on the various level of Java interviews and written tests, where you need to write code. On difficulty level, this question is at par with the prime numbers or Fibonacci series, which are also very popular on junior level Java programming interviews and it's expected from every programmer to know how to solve them.  
  
I personally like this exercise because it gives beginners an opportunity to familiarize themselves with the concept of Map data structure, which allows you to store mappings in the form of key and value.  
  
Since Map and Hash table data structure is heavily used in any enterprise Java application, good knowledge of this data structure is highly desirable among any level of Java programmers.  
  
By the way, there are a couple of variants of this problem, which you may want to look before going for an interview.  
  
Sometimes an interviewer will ask you to read a file and print all duplicate characters and their count, core logic will remain same, all you need to do is demonstrate how much you know about File IO in Java, like streaming file if it's very large rather than reading the whole file in memory.  
  
Btw, a basic knowledge of data structure and algorithms is needed and if you need to brush up then do so. If you need a resource, I highly recommend checking out Data Structures and Algorithms: Deep Dive Using Java course on Udemy. It's a hands-on course and covers all essential data structures. It's also very affordable and you can get in just $10 on Udemy flash sales which happens every now and then.

Java Program to find Repeated Characters of String

The standard way to solve this problem is to get the character array from String, iterate through that and build a Map with character and their count. Then iterate through that Map and print characters which have appeared more than once. So you actually need two loops to do the job, the first loop to build the map and second loop to print characters and counts.  
  
If you look at the below example, there is only one static method called printDuplicateCharacters(), which does both this job. We first got the character array from String by calling toCharArray().  
  
Next, we are using HashMap to store characters and their count. We use containsKey() method to check if key, which is a character already exists or not if already exists we get the old count from HashMap by calling get() method and store it back after incrementing it by 1.  
  
[](https://medium.com/javarevisited/top-10-free-data-structure-and-algorithms-courses-for-beginners-best-of-lot-ad807cc55f7a)Once we build our Map with each character and count, the next task is to loop through Map and check each entry, if count, which is the value of Entry is greater than 1, then that character has occurred more than once. You can now print duplicate characters or do whatever you want with them.

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

import java.util.Set;

/\*\*

\* Java Program to find duplicate characters in String.

\*

\*

\* @author http://java67.blogspot.com

\*/

public class FindDuplicateCharacters{

public static void main(String args[]) {

printDuplicateCharacters("Programming");

printDuplicateCharacters("Combination");

printDuplicateCharacters("Java");

}

/\*

\* Find all duplicate characters in a String and print each of them.

\*/

public static void printDuplicateCharacters(String word) {

char[] characters = word.toCharArray();

// build HashMap with character and number of times they appear in String

Map<Character, Integer> charMap = new HashMap<Character, Integer>();

for (Character ch : characters) {

if (charMap.containsKey(ch)) {

charMap.put(ch, charMap.get(ch) + 1);

} else {

charMap.put(ch, 1);

}

}

// Iterate through HashMap to print all duplicate characters of String

Set<Map.Entry<Character, Integer>> entrySet = charMap.entrySet();

System.out.printf("List of duplicate characters in String '%s' %n", word);

for (Map.Entry<Character, Integer> entry : entrySet) {

if (entry.getValue() > 1) {

System.out.printf("%s : %d %n", entry.getKey(), entry.getValue());

}

}

}

}

Output

List of duplicate characters in String 'Programming'

g : 2

r : 2

m : 2

List of duplicate characters in String 'Combination'

n : 2

o : 2

i : 2

List of duplicate characters in String 'Java'

Write a Program to Find Sum of Digits in Java

One of the common programming practice question thrown to beginners is to write a program to calculate the sum of digits in an integral number. For example, if the input is 123456 then output or sum of the digit is (1+2+3+4+5+6) = 21. An additional condition is you can not use any third party or library method to solve this problem. This program is not as simple as it looks and that's why it's a good exercise, you must know some basic programming techniques e.g. loops, operators, and logic formation to solve this problem. Let's see how we can solve this problem using Java programming language. In order to calculate the sum of digits, we must get digits as numbers. So your first challenge is how do you get the digits as numbers?  How do we extract 6 out of 123456?  
  
If you have done exercises like palindrome check or reversing number, then you should know that there is very old technique of getting last digit from a number by using modulus operator. If we do 123456%10 then we will get 6, which is last digit. In order to get all digits we can use a loop, something like while loop.  
  
Now our next challenge is how do we reduce number in each iteration so that our loop will finish as soon as we are done with all digits of number? Now coming from same palindrome problem, you can use technique of dividing number by 10 to get rid of last digit or reduce it by factor of 10.  
  
For example 123456/10 will give you 12345, which is one digit less than original number. So you got your end condition for while loop, check until number is not equal to zero. These two techniques are very important and can be used in variety of problem, so always remember these.

Java program to find Sum of Digits in Java

Here is our complete Java program to solve this problem. As explained in first paragraph, it does not use any library method instead uses division and modulus operator to calculate sum of digits of a number.

import java.io.Console;

import java.util.Scanner;

import java.util.concurrent.Semaphore;

import java.util.concurrent.locks.Condition;

import java.util.concurrent.locks.Lock;

import java.util.concurrent.locks.ReentrantLock;

/\*\*

\*

\* How to find sum of digits in Java

\*

\* @author Javin Paul

\*/

public class SumOfDigits{

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

System.out.println("Please enter a number to calculate sum of digits");

int number = sc.nextInt();

// Remember number/10 reduces one digit from number

// and number%10 gives you last digit

int sum = 0;

int input = number;

while (input != 0) {

int lastdigit = input % 10;

sum += lastdigit;

input /= 10;

}

System.out.printf("Sum of digits of number %d is %d", number, sum);

// closing Scanner to prevent resource leak

sc.close();

}

}

[](https://1.bp.blogspot.com/-VjiUpeuJ6ew/U6mChckLsTI/AAAAAAAABnA/EcxHGZFxClE/s1600/Java+Program+to+find+Sum+of+Digits.jpg)

Please enter a number to calculate sum of digits

101

Sum of digits of number 101 is 2

Please enter a number to calculate sum of digits

123

Sum of digits of number 123 is 6

That's all on how do you calculate sum of digits of a number in Java. It's an interesting exercise and you are welcome to find other solution as well. Try not to see the solution before doing it because if you can come up with logic by your own, you will learn a lot. These kind of programs are good for learning basic programming techniques and developing coding sense. If you are interested, you can find lot of such questions in this blog e.g. checkout this 10 programming questions article.

Java Program to Print Alphabets in Upper and Lower Case

One of the textbook exercise to get start with any programming language is writing a program to print alphabets in both upper and lower case. This program allows you to explore the String class in Java with toUpperCase() and toLowerCase() method but normally when you start, it's asked to do this without any API methods. This kind of exercise actually improves your understanding of programming language e.g. basic operators, data types like int and char. It's similar to your prime number, Fibonacci series, and factorial program exercise. I strongly suggest doing this textbook exercises to anyone who is just started learning a new programming language. Coming back to this program, Java has a datatype called char, which is 2-byte unsigned integer type. It is used to store characters in Java  e.g. char A = 'A'.  
  
Similar to String literal "A" we also have character literal 'A' which is letters enclosed in single quotes.  There is worth-noting difference between storing character literal in char and int data type in Java. If you store character literal on integer variable e.g. int i = 'a'; will store ASCII value of 'a' and when you print, it will prints ASCII value of 'a'.

How to print alphabets in upper and lower case

[Write a program to print alphabets in Java](https://3.bp.blogspot.com/-1lzFJzIgaHk/UF2Ci6kY5pI/AAAAAAAAAes/OYiM7r-DHzc/s1600/17.jpg)Here is our complete Java program to print characters in upper and lower case. It's simple program without using any API method. We have two methods printAlphabets() and printAlphabetsInUpperCase(), I have made them static method so that I can call them directly from main method, which is a static method. Why? because you can not call a non-static method from static context in Java.  
  
If you look at these methods, they are most simplest, you will ever see, of-course apart form HelloWorld in Java. Their is a loop in each of these method which print value of character in each iteration and runs until  it reaches last character of alphabets in each case.

/\*\*

\*

\* Java program to print alphabets in both upper and lower case.

\*

\* @author http://java67.blogspot.com

\*/

public class PrintAlphabetsInJava{

public static void main(String args[]) {

// printing alphabets in lower case i.e. 'a' to 'z'

printAlphabets();

// printing alphabets in upper case i.e. 'A' to 'Z'

printAlphabetsInUpperCase();

}

public static void printAlphabets() {

System.out.println("List of alphabets in lowercase :");

for (char ch = 'a'; ch <= 'z'; ch++) {

System.out.printf("%s ", ch);

}

}

public static void printAlphabetsInUpperCase() {

System.out.println("\nList of alphabets in upper case :");

for (char ch = 'A'; ch <= 'Z'; ch++) {

System.out.printf("%s ", ch);

}

}

}

Output

List of alphabets in lowercase :

a b c d e f g h i j k l m n o p q r s t u v w x y z

List of alphabets in upper case :

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

That's all on how to print alphabets on lower and upper case in Java. You can ever try to do this task in different way, try using different loop e.g. while, for-each or do-while. You can also try following programming exercise to get some more practice.