

[Home](#) » [Algorithm](#) » [Bit Manipulation](#) » [Interviews](#) » [Java](#) » Find all subsets of a set (Power Set)

## Find all subsets of a set (Power Set)

IN [ALGORITHM](#), [BIT MANIPULATION](#), [INTERVIEWS](#), [JAVA](#) - ON 13:26:00 - [NO COMMENTS](#)

Visit us:

**Print power set of any given set OR Print all subsets of a set OR Find all subsets of a set OR Print all subset of an array.**

Given a set of numbers, print all the possible subsets of it including empty set.

### What is Power Set?

In mathematics, the power set (or powerset) of any set  $S$ , written  $P(S)$ , is the set of all subsets of  $S$ , including the empty set and  $S$  itself.

Let's understand it with help of example.

Case 1:

Input: Set( $S$ ) = [1,2]

Output: PowerSet  $P(S)$  = [[ ], [2], [1], [2, 1]]

Case 2:

Input: Set( $S$ ) = [a,b,c]

Output: PowerSet  $P(S)$  = [[ ], [a], [b], [c], [a,b], [a, c], [b, c], [a, b, c]]

### Solution

There can be many approach to solve this problem, here we will look at 2 simple approaches.

1. Using Tree representation approach.
2. Using Bit manipulation approach.

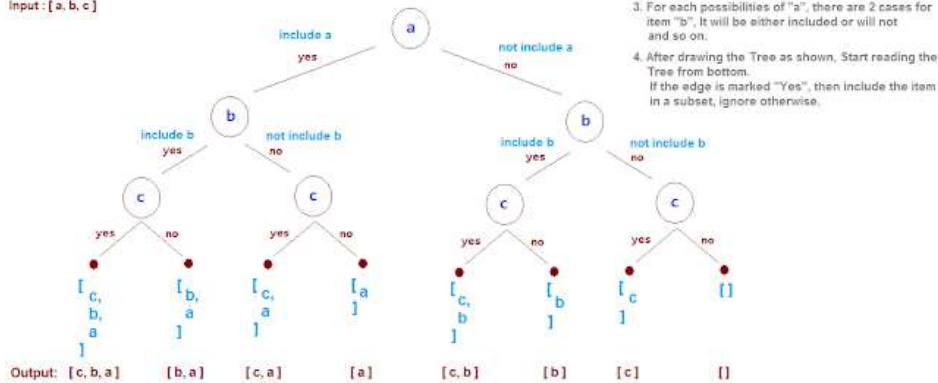
### Tree Representation Approach

In this approach, we will draw a Tree like shown below,



## Find all subsets of a given set

Input: [a, b, c]



There are 2 possibilities for each item of a set, It will be either included in a subset or not included.

Based on this fact, **Lets draw a Tree for given Set [a, b, c].**  
**Visit us:**

There are 2 possibilities of "a". It will be either included in a subset or not included.

Similarly, For each possibilities of "a", again there are 2 possibilities for "b".

"b" will be included for each possibilities of "a" or not included.

Similarly, For each possibilities of "b", again there are 2 possibilities for "c".

Again, For "c", there are 2 possibilities either it will be considered or will be ignored in a subset.

**Stop as there are no more items in a set remaining. Our Tree is complete and will look as shown in image above.**

Lets understand our example and draw a Tree for it.

**Set [a, b, c].**

**"a" is included.**

1. "b" will be included in a subset.
  2. "c" will be included in a subset.      END- No more element in Set.
  2. "c" will not be included in a subset.      END- No more element in Set.
1. "b" will not be included in a subset.
  2. "c" will be included in a subset.      END- No more element in Set.
  2. "c" will not be included in a subset.      END- No more element in Set.

**"a" is not included.**

1. "b" will be included in a subset.
  2. "c" will be included in a subset.      END- No more element in Set.
  2. "c" will not be included in a subset.      END- No more element in Set.
1. "b" will not be included in a subset.
  2. "c" will be included in a subset.      END- No more element in Set.
  2. "c" will not be included in a subset.      END- No more element in Set.

**Now, read the Tree from Bottom to Root. Wherever there is Yes, include the item in a subset, ignore otherwise.**

**"a" is included.**

1. "b" will be included in a subset.
  2. "c" will be included in a subset.      END- No more element in Set. = [ c, b, a ]
  2. "c" will not be included in a subset.      END- No more element in Set. = [ b, a ]

1. "b" will not be included in a subset.
2. "c" will be included in a subset.      END- No more element in Set. = [ c, a ]
2. "c" will not be included in a subset.      END- No more element in Set. = [ a ]

"a" is not included.

1. "b" will be included in a subset.
2. "c" will be included in a subset.      END- No more element in Set. = [ c, b ]
2. "c" will not be included in a subset.      END- No more element in Set. = [ b ]

1. "b" will not be included in a subset.
2. "c" will be included in a subset.      END- No more element in Set. = [ c ]
2. "c" will not be included in a subset.      END- No more element in Set. = [ ]

Subset or PowerSet of [a, b, c] is [ [ c, b, a ], [ b, a ], [ c, a ], [ a ], [ c, b ], [ b ], [ c ], [ ] ]

Visit us:

## Understanding Program

Algorithm we will use for printing subset of a set is,

Step 1. Collect empty subset ( [ ] ) first because empty subset will be subset of any set including null set.

Step 2. Now, For each item of a Set, Iterate subsets already found and

1. Include a item in each subset and
2. Not include a item in each subset.

Set S = [a, b, c] ➡

```

Add empty set to subset list first.
Subsets = [ [ ] ]
i = a, b, c
For a,
  Subsets = [ ]
  Include "a" in [ ].      ([a])
  Not include "a" in [ ]  ([ ])      Subsets = [ [ ], [a] ]

For b,
  Subsets = [ ], [a]
  Include "b" in [ ].      ([b])
  Not include "b" in [ ]  ([ ])
  Include "b" in [a].      ([b, a])
  Not include "b" in [a]  ([a])      Subsets = [ [b], [ ], [b, a], [a] ]

For c,
  Subsets = [b], [ ], [b, a], [a]
  Include "c" in [b].      ([c, b])
  Not include "c" in [b]  ([b])
  Include "c" in [ ].      ([c])
  Not include "c" in [ ]  ([ ])
  Include "c" in [b,a].    ([c, b, a])
  Not include "c" in [b,a] ([b, a])
  Include "c" in [a].      ([c, a])
  Not include "c" in [a]  ([a])      Subsets = [ [c, b], [b], [c], [ ], [c, b, a], [b, a], [c, a], [a] ]

```

## Java Program for printing Subsets of set using Tree representation approach.

```

1 package string;
2
3 import java.util.ArrayList;
4 import java.util.Iterator;
5 import java.util.List;
6
7 public class FindAllSubsetOfSet {
8
9     public static void main(String[] args) {
10         System.out.println("\n"+createSubsetUsingTree("A"));
11     }
12
13     private static List<String> createSubsetUsingTree(String str){
14

```

```

15 List<String> result = new ArrayList<String>(); // take set if you want unique res
16 result.add("");
17
18 //If str is not null, then process further otherwise return empty set.
19 if(str!=null && str.length()>0){
20
21     //Iterate each element of a set
22     for (int i = 0; i < str.length(); i++) {
23
24         //Working on str.charAt(i);
25         //Store the result of subset of str.charAt(i) in tempList.
26         List<String> tempList = new ArrayList<String>();
27
28         //Add str.charAt(i) in each item of result.
29         Iterator<String> iter = result.iterator();
30         while(iter.hasNext()){
31             String val = iter.next();
32
33             // If val is [], it means str.charAt(i) is not included, So include it in resu
34             if(val.equals("")){
35                 tempList.add("[" + str.charAt(i) + "]");
36             }else{
37
38                 //For each item, there will be 2 subset, one including it and one without inc
39                 //If val is not [], it means it already contain some subset without str.charA
40                 tempList.add("[" + val.substring(1,val.length()-1) + ", " + str.charAt(i) + '
41             }
42         }
43
44         //Add all subsets present in tempList to final result.
45         result.addAll(tempList);
46     }
47 }
48
49 return result;
50 }
51
52 }

```

Visit 48:

## Bit Manipulation Approach

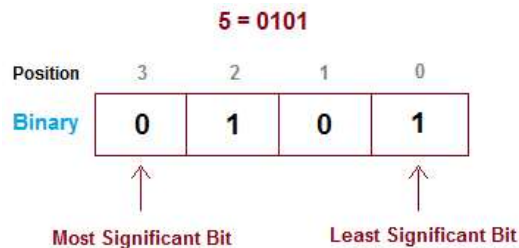
In this approach we will use Bits Manipulation for printing Subset of a set.

Before going ahead, first lets understand how to check any bit is set in a byte.

Binary representation of a number 1 = 0001, 2 = 0010, ..., 5 = 0101, ...

How to identify whether LSB, that is bit at position 0 is set in 5(0101)?

**Solution:**



Do a Logical AND of 1 and 5 that is (0001 & 0101).

If the result is greater than 0, then LSB, that is bit at position 0 is set in value 5.

How to identify whether bit at position 1 is set in 5(0101)?

**Solution:**

Do a Logical AND of 2 and 5 that is (0010 & 0101).

If the result is greater than 0, then bit at position 1 is set

So, to generalize it, if we want to check nth bit is set in a number,

**Step 1.** We will first take binary representation of 1 (0001),

**Step 2.** Move the LSB in binary representation of 1 to nth position.

Eg: If we want to check whether bit at position 2 is set, then move 1 two times on right side,

=  $1 \ll 2$

=  $0001 \ll 2$

= 0100

**Step 3.** Once the LSB of binary 1 is moved to nth position, which we want to check,

just do logical AND of binary representation of 1 with the number,

If the result is greater than 0, then the bit is set, not otherwise.

**If we want to find subset of 3 characters [a, b, c], then how many total subset can be made out of it?**

If you observe, the number of subsets equals to 2 to the power of the number of elements in the set.

Number of subsets that can be made from set of 3 characters [a, b, c] is  $2^3 = 8$  subsets.

Number of subsets that can be made from set of 4 characters [a, b, c, d] is  $2^4 = 16$  subsets.

**Visit us:**

So, if there are n characters [a, b, c, ..., n] in a set, then there will be  $2^n$  total subsets.

**How to find total number of subsets possible using bit manipulation?**

By shifting the bit representation of the number 1 by n, we will get  $2^n$ .

Thus, if we shift the bit string of 1 by the number of elements in the given set, we'll get the number of subsets for that set.

For example, if we have  $S = [a, b, c]$ , length of set here is 3,

So, for finding total subsets, we have to shift binary 1, 3 times,

$1 \ll 3 = (0001 \ll 3) = (1000) = 2^3 = 8$  subsets of set S.

If we compare subset of [a, b, c] to binaries representation of numbers from 0 to 7, we can find a relation with the bit sets in each number to subsets of [a, b, c].

Number	Binary representation	Subset
0	000	= Empty set
1	001	= c
2	010	= b
3	011	= bc
4	100	= a
5	101	= ac
6	110	= ab
7	111	= abc

**Map 3 positions of Binary Representation to 3 characters of a set [a, b, c]**

Binary representation position maps to Position of a characters in a Set

Pos1	Pos2	Pos3
------	------	------



Pos1	Pos2	Pos3
------	------	------

Example:

0	0	1
---	---	---



		c
--	--	---

0	1	0
---	---	---



	b	
--	---	--

1	1	1
---	---	---



a	b	c
---	---	---

## Understanding Program

length of a Set = 3  
Total Subsets =  $2^3 = 8$  Subsets:

**i = 0 to 7**

For 0 = 000  
check each bit at position 0, 1 and 2 in 000.  
If bit at position 0 is set, then "a" will be part of subset.      Subset = {[ ]}  
If bit at position 0 is not set, then "a" will not be part of subset.  
  
If bit at position 1 is set, then "b" will be part of subset.  
If bit at position 1 is not set, then "b" will not be part of subset.  
  
If bit at position 2 is set, then "c" will be part of subset.  
If bit at position 2 is not set, then "c" will not be part of subset.

Set S= [a, b, c] →

For 1 = 001  
check each bit at position 0, 1 and 2 in 001.  
  
Bit at position 0 is set, NO, then "a" will not be part of subset.  
Bit at position 1 is set, NO, then "b" will not be part of subset.  
Bit at position 2 is set, YES, then "c" will be part of subset.      Subset = {[ ], [c]}

⋮

For 7 = 111  
check each bit at position 0, 1 and 2 in 111.  
  
Bit at position 0 is set, YES, then "a" will be part of subset.  
Bit at position 1 is set, YES, then "b" will be part of subset.  
Bit at position 2 is set, YES, then "c" will be part of subset.      Subset = {[ ], [a,b,c]}

Visit us:

## Java Program for printing Subsets of set using Bit Manipulation approach.

```
1 package string;
2
3 import java.util.ArrayList;
4 import java.util.HashSet;
5 import java.util.List;
6 import java.util.Set;
7
8 public class FindAllSubsetOfSet {
9
10     public static void main(String[] args) {
11         List<Object> list = new ArrayList<Object>();
12         list.add("a");
13         list.add("b");
14         list.add("c");
15
16         System.out.println(getSubsetUsingBitMap(list));
17     }
18
19     private static Set<Set<Object>> getSubsetUsingBitMap(List<Object> list){
20
21         Set<Set<Object>> result = new HashSet<Set<Object>>();
22
23         int numOfSubsets = 1 << list.size(); //OR Math.pow(2, list.size())
24
25         // For i from 0 to 7 in case of [a, b, c],
26         // we will pick 0(0,0,0) and check each bits to see any bit is set,
27         // If set then element at corresponding position in a given Set need to be included
28         for(int i = 0; i < numOfSubsets; i++){
29
30             Set<Object> subset = new HashSet<Object>();
31
32             int mask = 1; // we will use this mask to check any bit is set in binary representation
33
34             for(int k = 0; k < list.size(); k++){
35
36                 if((mask & i) != 0){ // If result is !=0 (or >0) then bit is set.
37                     subset.add(list.get(k)); // include the corresponding element from a given set
38                 }
39
40                 // check next bit in i.
41                 mask = mask << 1;
42             }
43
44             // add all subsets in final result.
45             result.add(subset);
46         }
47         return result;
48     }
49
50 }
```

## You may also like to see

[Write a program to print all permutations of a given string without repetition. \(Repetition of characters is not allowed\).](#)

[Write a program to print all permutations of a given string with repetition. \(Repetition of characters is allowed\).](#)

[Print all combinations/permutations of a string of length k that can be formed from a set of n characters](#)

[How ConcurrentHashMap works and ConcurrentHashMap interview questions](#)

Visit us:

[How Much Water Can A Bar Graph with different heights can Hold](#)

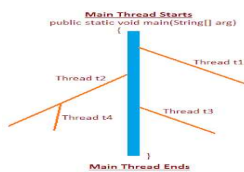
[Interview Questions-Answer Bank](#)

Enjoy !!!!

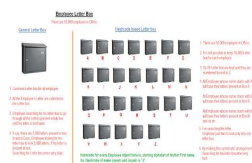
If you find any issue in post or face any error while implementing, Please comment.



## RELATED POSTS :



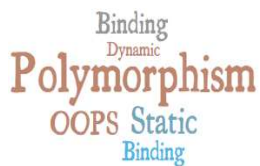
What is Thread in Java with example...



How Hashmap works in Java



Quartz Scheduler Tutorial In Java w...



Static Polymorphism in Java with Pr...

LABELS: [ALGORITHM](#), [BIT MANIPULATION](#), [INTERVIEWS](#), [JAVA](#)

9 Comments Java By Patel

[Login](#)

[Recommend 1](#) [Share](#)

[Sort by Best](#)



Join the discussion...

LOG IN WITH

OR SIGN UP WITH EMAIL

LOG IN WITH

OR SIGN UP WITH DISQUS [?](#)



**AnOop PooNia** • a year ago

best explanation ... god bless u brother

^ | v • Reply • Share ›



**Vaibhavi** • a year ago

Most thorough and original explanation of Power set!

Thank you for your efforts.

^ | v • Reply • Share ›



**Ankita** • a year ago

Thank you for the post !!

2 questions:

1) will the time complexity of tree method be  $O(2^N)$

2) how to handle duplicates in both methods ? I tried using boolean hashset but failed

Thanks in advance.

^ | v • Reply • Share ›



**Venkat Pavan** • a year ago

best explanation even a newbie like me understood the logic

^ | v • Reply • Share ›



**Jayesh Patel** Mod → Venkat Pavan • a year ago

Thanks Venkat...

^ | v • Reply • Share ›



**dogbert82** • 2 years ago

Thanks for this very detailed explanation! I think this is the best ever explanation for subset problem I have ever seen! Once again thank you....I really enjoyed it..

^ | v • Reply • Share ›



**Jayesh Patel** Mod → dogbert82 • 2 years ago

Thanks dogbert82....Your kind words means a lot to me and I am glad it helped..

^ | v • Reply • Share ›



**M.I.** • 2 years ago

These lines are not very clear mate.

"If the result is greater than 0, then LSB, that is bit at position 0 is set in value 5."

and

"If the result is greater than 0, then bit at position 1 is set"

Thanks.

^ | v • Reply • Share ›



**Jayesh Patel** Mod → M.I. • 2 years ago

First of all sorry for late reply. I can understand, my words are confusing. I will rework on this and explain it in better way. Many thanks for feedback.

^ | v • Reply • Share ›

ALSO ON JAVA BY PATEL

**How to run two thread one after another in java**

1 comment • a year ago

[Bulky Rajend](#) • what is the point of looping till

**Print Matrix Diagonally OR Diagonal order of Matrix.**

4 comments • 2 years ago

[shrestha](#) • hi!! Please suggest me why the

Visit us:



**Rishi Rajput** — what is the point of looping till 100 in this example ?

### Serialization Interview Questions In Java

2 comments • a year ago

**Jayesh Patel** — Hi Sukhendnu, Thanks for writing to me. Please find link to detailed post on this below,<http://javabypatel.blogspot...>

**viral tyagi** — niii, Please suggest me why the above concept gives does not work, it gives me following exception. E:\java practice\Array ...

### Why default constructor is not called during Deserialization process in Java.

2 comments • a year ago

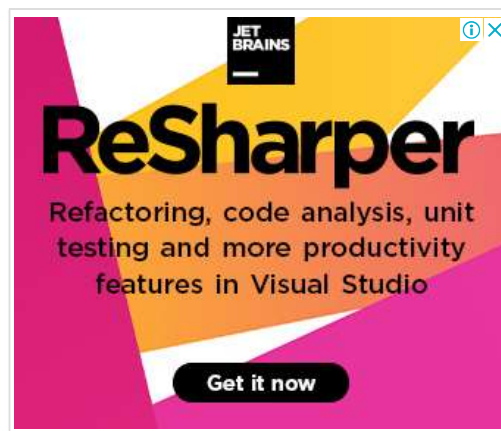
**Jayesh Patel** — Thanks Jitendra.

[Newer Post](#)

[Home](#)

[Older Post](#)

Visit us:



*I'm Jayesh Patel, author of "JavaByPatel".*

*I'm not a professional blogger but when time permits, love to share in-depth solutions to common Interview questions asked.*

*Any questions/feedback, Please drop a mail at [jayeshmaheshpatel@gmail.com](mailto:jayeshmaheshpatel@gmail.com)*



#### CATEGORIES

Algorithm    Angular2    AngularJS  
Array    Backtracking  
Binary Search Tree    Binary Tree  
Bit Manipulation    Database  
Datastructure    Design Pattern  
Graph    Interviews    Java    Java8  
Javascript    Linked List    Matrix  
Miscellaneous    Multithreading  
OOPS    Puzzle    Quartz    Queue  
Sorting    Spring    Stack    Stream  
Strings    Typescript    Web Service

Visit us:

## Windows

### POPULAR POSTS



**Convert integers to roman numerals equivalent in Java**  
Converting Integers to Roman Numerals equivalent in Java ...



**How time complexity of Hashmap get() and put() operation is  $O(1)$ ? Is it  $O(1)$  in any condition?**  
How time complexity of Hashmap get() and put() operation...



**What is Load factor and Rehashing in Hashmap?**  
What is Load factor and Rehashing in Hashmap? This is...



**How ConcurrentHashMap works and ConcurrentHashMap interview questions.**  
ConcurrentHashMap Interview Questions In Java. ...



**Find all subsets of a set (Power Set)**  
Print power set of any given set OR Print all subsets of...

### RECENT COMMENTS

READY, SET, BUILD!

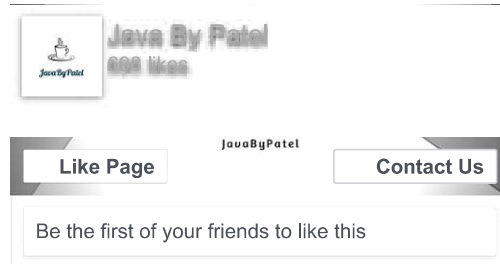
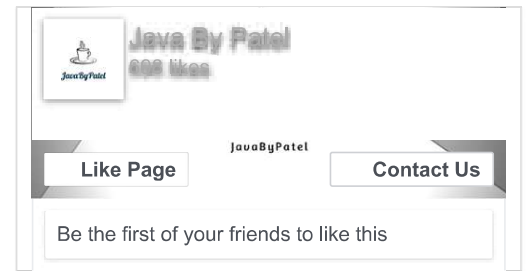
# TeamCity

HASSLE-FREE CI AND CD SERVER

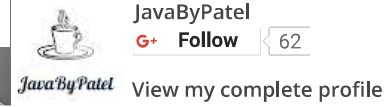
[Try free now](#)

JET BRAINS

Visit us:



FOLLOW ME ON GOOGLE+



Powered by Blogger.

#### BLOG ARCHIVE

- › 2013 (2)
- › 2017 (102)
- › 2016 (63)
- › 2015 (48)
  - 🕒 2015 December (7)
  - 🕒 2015 November (3)
  - 🕒 2015 October (9)
    - 🕒 Check If two strings are Anagrams.
    - 🕒 Find all subsets of a set (Power Set)
    - 🕒 Given two words (beginWord and endWord), and a dic...
    - 🕒 Print Nodes in Bottom View of Binary Tree.
    - 🕒 Print Nodes in Top View of Binary Tree
    - 🕒 Print a Binary Tree in Vertical Order. Find Vertic...
    - 🕒 What is Load factor and Rehashing in Hashmap?
    - 🕒 How time complexity of Hashmap get() and put() ope...

🕒 How HashMap data structure works internally? How h...

🕒 📅 September (10)

🕒 📅 August (12)

🕒 📅 July (7)

---

© COPYRIGHT 2013 JAVABYPATEL - ALL RIGHTS RESERVED - POWERED BY BLOGGER.COM



**Visit us:**