

Please complete the textbook section 3.1 before beginning this assignment.

CS 278 / 465

PA 5: Forming Subsets

All programs you submit in this class must follow the Documentation and Style Guidelines.
This document can be found in the Canvas Modules.

All programs you submit in this class must compile with the Oracle Java compiler on the
Linux machines in SH 118 or SH 118B.

A **set** is a collection of *distinct* elements in which the order of the elements is not important.

So $\{1, 2, 3\}$, $\{3, 4\}$, $\{\}$ and $\{5, 99, -1\}$ are all sets.

Because the order of the elements is not important, $\{1, 2, 3\}$ and $\{3, 2, 1\}$ are the same set.

Because your brain works better when there is some order, we'll focus on sets of the form

$\{1, 2, \dots, n\}$. For example, $\{1, 2, 3\}$. (Always keep in mind, though, that sets can contain anything as elements.)

The set S_1 is said to be the **subset** of the set S_2 , if all the elements of S_1 also belong to S_2 . The empty set is a subset of every set.

Knowing this, it's easy to figure out the subsets of $\{1, 2, 3\}$:

$\{\}$
 $\{1\}$
 $\{2\}$
 $\{1, 2\}$
 $\{3\}$
 $\{1, 3\}$
 $\{2, 3\}$
 $\{1, 2, 3\}$

How many subsets are there? _____

A set with 1 element has two subsets: $\{\}$ and $\{1\}$.

For a set of 2 elements, there are 4 subsets: $\{\}$, $\{1\}$, $\{2\}$, $\{1, 2\}$.

For a set of 3 elements, there are 8 subsets. Notice the pattern?

$n = 1$: 2^1 subsets
 $n = 2$: 2^2 subsets
 $n = 3$: 2^3 subsets
 $n = n$: 2^n subsets

Note that the explanation below has been written with Java in mind. Java has a boolean primitive type.

It's easy to figure out how many subsets there will be. But, it's not so easy to generate the subsets. There are many algorithms for doing this, but we'll be focusing on just one.

The algorithm we'll use involves using a **mask**. (In a program, a set is stored in an array and the mask is stored in a second array.)

If you have a set of n elements, a valid mask would be an array of n boolean elements. When you apply a mask to a set, you check each element in the set and the corresponding one in the mask. If `mask[i]` is true, you add `set[i]` to the subset, otherwise, you ignore it.

For example, after applying the mask `[false, true, false, false, true]` to the set `{1, 2, 3, 4, 5}`, you get the subset `{2, 5}`.

So, to generate all the subsets of a set of n elements, you first have to generate all the possible 2^n masks of the set and then apply them.

The Big Picture

In this assignment, you will write a computer program to solve the following problem. Given a set of integers S and an integer n , find all non-empty subsets of S whose sum is n .

For example, given the set $S = \{1, 2, -1, 3\}$ and the number $n = 3$, there are 3 subsets of S whose sum is 3. The subsets that meet the criteria are $\{1, 2\}$, $\{1, -1, 3\}$, and $\{3\}$.

Now for the Details

1. Create a new folder for PA5. Download the file `Masks.java` and store it in the PA5 folder.
2. Run the `Masks.java` program a few times. The input is 1) the size of the set and 2) the elements of the set.

This program generates and prints all subsets of a given set of integers. Verify that the output is correct.

Study the program code intently. DO NOT MODIFY THIS PROGRAM.

The set is stored in an array of integers. A mask is stored in an array of booleans.

The subsets are not stored anywhere.

There is a main method and two other methods. The other methods are:

a) `next`

parameter: an array of booleans called `mask`

purpose: modify the boolean array so that it contains the pattern for the next mask

return value: boolean, true if it successfully generated the next mask, false if it could not generate another mask.

b) `printSubset`

parameters: an array of integers (the set) and an array of booleans (the mask)

purpose: build and print a string representation of a subset

Note: the method does not declare any additional arrays and does not save the subset

return value: none

3. Create a new file called `PA5.java`. Copy and paste the code from `Masks.java` into `PA5.java`. You can copy the entire program (then remove unneeded pieces), or just copy the pieces that you need.
4. Modify the code or write new code so that your program solves the problem given in "The Big Picture". Here are the major tasks:

a) input the size of the set `S`

b) input the elements of the set `S`

c) input the number `n`

d) print only the subsets of `S` whose sum is `n`

Word of Advice

Don't modify the next method.

Submit `PA5.java` on Canvas.