House of cats explanation

Problem

There are some people and cats in a house. I am given the number of legs they have all together. My task is to return an array containing every possible number of people that could be in the house sorted in ascending order. It's guaranteed that each person has 2 legs and each cat has 4 legs.

Example:

For legs = 6, the output should be

houseOfCats(legs) = [1, 3].

There could be either 1 cat and 1 person (4 + 2 = 6) or 3 people (2 \* 3 = 6).

For legs = 2, the output should be

houseOfCats(legs) = [1].

There can be only 1 person.

Input/Output

[input] integer legs

The total number of legs in the house. It's guaranteed, that this number is even.

Guaranteed constraints:

0 ≤ legs < 50.

[output] array.integer

Every possible number of people that can be in the house.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Firstly, I create a class MainClass. In this assignment, I have to enter in the console the total number of the legs in the house. This is why in my class I have a Scanner (in order to use it I have to add java.util.Scanner).

Scanner input = **new** Scanner(System.***in***);

**int** numberOfLegs=0;

I create an object of type Scanner and a variable of type integer numberOfLegs and give it an initial value 0, because afterwards if there is any exception or any problem with the input, I’m going to return zero. I have do-while loop, because I want to execute it once, before checking if the conditions are true. Inside the loop I created a try-catch block that checks if the entered value is a valid number (is it numeric) and if it is not, it prints a message. input.nextLine() is for clearing the invalid input from the scanner’s buffer.

**do** {

System.***out***.println("Please, enter a valid even number between 0 and 50:");

**try** {

numberOfLegs = input.nextShort();

}

**catch** (InputMismatchException e) {

System.***out***.println("Not a valid number! Please, try again.");

input.nextLine();

}

}**while**(numberOfLegs<0 || numberOfLegs>50 || numberOfLegs%2!=0 );

Then, I print the entered value (the total legs in the house).

I created two integer variables, one for the count of the cats and one for the count of the people in the house.

**int** countCats = 0;

**int** countPeople = 0;

Using the for loop, I found the total number of the cats and the total number of the people (I increase the i value +4 times, because cats have 4 legs, and +2 for the people (2 legs)).

**for**(**int** i=0;i<numberOfLegs;i+=4) {

countCats++;

}

**for**(**int** i=0;i<numberOfLegs;i+=2) {

countPeople++;

}

Then, I don’t know how much elements I am going to have in the array, so I created a List class, because it’s resizable (in order to use it I added

**import** java.util.List;

**import** java.util.ArrayList;). As a first element I added the maximum possible number of people and then I nullified the countPeople variable, because I’m going to assign new values to it.

List<Integer> listOfNumbers=**new** ArrayList<>();

listOfNumbers.add(countPeople);

countPeople = 0;

In the first outer for loop, i<=countCats, because the loop has to start over again as much as the total number of the cats. In the inner for, I check the j value and it starts from zero and increases with two, because people have two legs, j<numberOfLegs-(i\*4) the loop will start over again that much times, for example first time, it will count the total people when there is only one cat in the house, second time, when there is two cats and etc. When I found the count, I add it to the list and again assign 0 to the variable so I can find the other possible numbers of people.

**for**(**int** i=1;i<=countCats**;** i++) {

**for**(**int** j=0;j<numberOfLegs-(i\*4);j+=2) {

countPeople++;

}

listOfNumbers.add(countPeople);

countPeople=0;

}

Then, I converted the list to an array and using the arrays method of sorting (in order to use it I imported java.util.Arrays) sorted the array. Finally is the output of the elements of array, using the for loop again. There is an if condition (n!=array.length-1 && array[n]!=0) in order not to print a comma after the last element and not to print zeros as an element.

Integer[] array = listOfNumbers.toArray(**new** Integer[0]);

Arrays.*sort*(array);

System.***out***.print("houseOfCats(legs) = [");

**for**(**int** n=0;n<array.length;n++) {

**if**(n!=array.length-1 && array[n]!=0)System.***out***.print(array[n]+",");

**if**(n==array.length-1) System.***out***.print(array[n]+"]");

}