**Most Valuable Paths**

**Overview**

Your task is to implement a program, which receives a tree-like structure of integers and find the top 3 most valuable paths you could walk through it… Or in other words – the paths with the highest summed up value.

**Input**

* You will begin receiving lines of input – each containing 2 integers separated by a space. You need to generate the tree from there.
* The input ends when you receive the command "**calculate**".

**Output**

* As output you need to print the top 3 paths in terms of summed up value and printed them in descending order of their sum, in the following format:
  + **{pathSum} -> {pathNode1} {pathNode2}…**

**Constraints**

* The given numbers will be valid integers in range [-231, 231 – 1].
* Nodes will always be unique.
* There will always be at least 2 nodes.
* There will be no paths with equal values.

**Examples**

| **Input** | **Output** | **Comment** |
| --- | --- | --- |
| 7 19  7 23  7 14  19 6  19 9  14 41  14 8  calculate | 62 -> 7, 14, 41  35 -> 7, 19, 9  32 -> 7, 19, 6 |  |
| 15 10  15 5  15 20  15 25  10 6  6 31  31 4  31 7  10 21  21 16  5 45  45 0  45 9  20 38  20 37  38 41  25 22  22 19  calculate | 114 -> 15, 20, 38, 41  81 -> 15, 25, 22, 19  74 -> 15, 5, 45, 9 |  |