Problem:
Assume that a robot is being built that will grab toys from a bin, and hand them to kids. You need to implement a component of the software part (not the physical robot).
The 'presents' are given as strings: each present is a string with the present description. E.g.: "xylophone", "car", "doll house", "Lego set 1", "sword".

• **Version 1** (easiest): Assume there are two kids: Bob and Alice. The robot starts by giving the first present to Alice and then the next one to Bob and then to Alice and so on. It takes turns handing out presents to each kid. Print what presents each kid gets. (E.g.: Bob gets: ……. Alice gets: ……..)

• **Version 2**: Give the presents ‘at random’ to the two kids (flip a coin).
  – What from the solution for version 1 can be kept and what needs to change?
  – Would you benefit if you change how you represent the data?

• **Version 3**: Generalize to any number, K, of kids.

• **Version 4 (in Homework)**: There is a label (A/B) with the present: "B car", "A doll house“ indicating who the present goes to (Alice/Bob)
Print the result **or store** it in variables?
To understand data and the process:
- add the indexes for the gifts array and
- Fill in the result data for Alice and Bob.
- Extend you thinking: where do X, Y and Z go?

<table>
<thead>
<tr>
<th>gifts</th>
<th>&quot;Doll&quot;</th>
<th>Drum</th>
<th>Paint</th>
<th>ball</th>
<th>Book</th>
<th>train</th>
<th>......</th>
<th>X</th>
<th>Y</th>
<th>...</th>
<th>Z</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>String[]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Draw and understand the data in your code; do NOT hardcode.
2. Print the result **or store** it in variables?
3. Process needed to produce the result data
   - Loop:
     • how many iterations? (init?, cond?, update?)
     • what does one iteration do? (there are at least 2 options)
     • Data (new or updates) needed to achieve it
     • Pseudocode and trace on paper
4. Refine and write code
   • Declare and init variables, think about size of arrays,
   • Special cases/edge cases