FIP ⇒ Array ; Integer

- Splits Array into 2 → Find Return Times (matching Number)
  - → Not Find (-1)

Array { empty
full → Mid = MAX / 2
→ Looping till (-1) Not find

- GOALS:

✓ - KEEP NOTE OF ERRORS (DURING ENCOUNTERING)

✓ - BINARY CHOP

✓ - TOTAL TESTS, MISSING ELEMENTS (WITH POSITION)

✓ - EVERY PASS DIVIDE INTO TWO → MODIFIED → DIRECT GETS THE MULTIPLE OCCURANCE

- SOLUTION

✓ INPUTS: (I) datasize
(II) missing (any random num)

✓ Output: (I) missingElem (gives missing num)
(II) All Numbers (Format: (Number: found in X times)
(III) Time of occurence

✓ Values: (I) RandomElementsIndex (use math.floor → random)
(II) cut = len/2 (Took roundUP)
(III) textVal = index of hay (hay[cut])
(IV) compare < needle, textVal
└ 0 ⇒ found → TestResults
└ 1 ⇒ Not found

(II) Print ⇒ TestTime + MissingElement + TestResults
(for online)