

```

    public static String unknown (int [] arr) {
        int k=0;
        int i = arr.length;
        if (i < 2) {
            return "inco";
        }
        else {
            while (i > 1) {
                k++;
                i--;
            }
            int subarr[] = arr.copyOfRange (arr, 0,
                arr.length/2);
            return k + "-" + unknown (subarr);
        }
    }
  
```

$$\textcircled{1} \quad T(n) = \begin{cases} cte_0 & ; n < 2 \\ cte_1 + \frac{n}{2} \cdot C_1 + T(n/2) & ; n \geq 2 \end{cases}$$

$$\textcircled{2} \quad T(n) = cte_1 + \frac{n}{2} \cdot C_1 + T(n/2) ; n \geq 2 ; i=1$$

$$\hookrightarrow T(n/2) = cte_1 + \frac{n/2}{2} \cdot C_1 + T(\frac{n/2}{2})$$

$$T(n) = cte_1 + \frac{n}{2} \cdot C_1 + cte_1 + \frac{n}{4} \cdot C_1 + T(\frac{n}{4})$$

$$T(n) = 2cte_1 + \frac{3n}{4} \cdot C_1 + T(\frac{n}{4}) ; n > 4 ; i=2$$

$$\hookrightarrow T(n) = cte_1 + \frac{n/4}{2} \cdot C_1 + T(\frac{n/4}{2})$$

$$T(n) = 2CTE_1 + \frac{3}{4}nC_1 + CTE_1 + \frac{n}{8}C_1 + T\left(\frac{n}{8}\right)$$

$$T(n) = 3CTE_1 + \frac{7}{8}nC_1 + T\left(\frac{n}{8}\right); n \geq 8; l=3$$

$$(3) T(n) = K CTE_1 + \frac{n}{2^k} C_1 + T\left(\frac{n}{2^k}\right); n \geq 2^k$$

$$(4) \frac{n}{2^k} = 1 \rightarrow n = 2^k \rightarrow k = \log_2(n)$$

$$(4) T(n) = \log_2(n) \cdot CTE_1 + 1 \cdot C_1 + T(1)$$

$$T(n) = \log_2(n) \cdot CTE_1 + 1 \cdot C_1 + CTE_1$$

$$T(n) = \log_2(n) \cdot CTE_1 + CTE_3; n \geq 1; CTE_3 = C_1 + CTE_0$$

$$(3) T(n) = K CTE_1 + nC_1 \cdot \sum_{i=1}^k \frac{1}{2^i} + T\left(\frac{n}{2^k}\right); n \geq 2^k$$

$$\frac{n}{2^k} = 1 \rightarrow n = 2^k \rightarrow k = \log_2(n)$$

$$(4) T(n) = \log_2(n) \cdot CTE_1 + n \cdot C_1 \cdot \sum_{i=1}^{\log_2 n} \frac{1}{n} + T(1)$$

$$T(n) = \log_2(n) \cdot CTE_1 + n \cdot C_1 \cdot \left(1 \cdot \frac{1}{n}\right) + T(1)$$

$$T(n) = \log_2(n) \cdot CTE_1 + C_1 \cdot (n-1) + CTE_0$$

$$O(n) = \alpha(\log(n)) \cdot O(n) \rightarrow O(n) = n \cdot \log(n)$$