Design and Analysis of Algorithms

03-02
Divide – and – Conquer

Linear Search

Imran Ihsan
Assistant Professor, Department of Computer Science
Air University, Islamabad, Pakistan
www.imranihsan.com

Linear Search in Array

Ann	Pat	…	Joe	Bob

Linear Search in Array

Linear Search in Array

Linear Search in Array

Linear Search in Array

Real Life Example

English	French	Italian	German	Spanish
House	Maison	Casa	Haus	Casa
Car	Voiture	Auto	Auto	Auto
Table	Table	Tavola	Tabelle	Mesa

Searching in an Array

Input: An array `A` with `n` elements.
A key `k`.

Output: An index, `i`, where `A[i] = k`.
If there is no such `i`, then `NOT_FOUND`.

Recursive Solution

```
LinearSearch(A, low, high, key)
      if high < low:
         return NOT_FOUND
      if A[low] = key:
         return low
      return LinearSearch(A, low + 1, high, key)
```

Definition

A recurrence relation is an equation recursively defining a sequence of values

Fibonacci Recurrence Relation

$$F_n = \begin{cases} 0, & n = 0\,, \\ 1, & n = 1\,, \\ F_{n-1} + F_{n-2}, & n > 1\,. \end{cases}$$

Recursive Solution

```
LinearSearch(A, low, high, key)
      if high < low:
         return NOT_FOUND
      if A[low] = key:
         return low
      return LinearSearch(A, low + 1, high, key)
```

Recurrence defining worst-case time:

$$T(n) = T(n - 1) + c$$

Recursive Solution

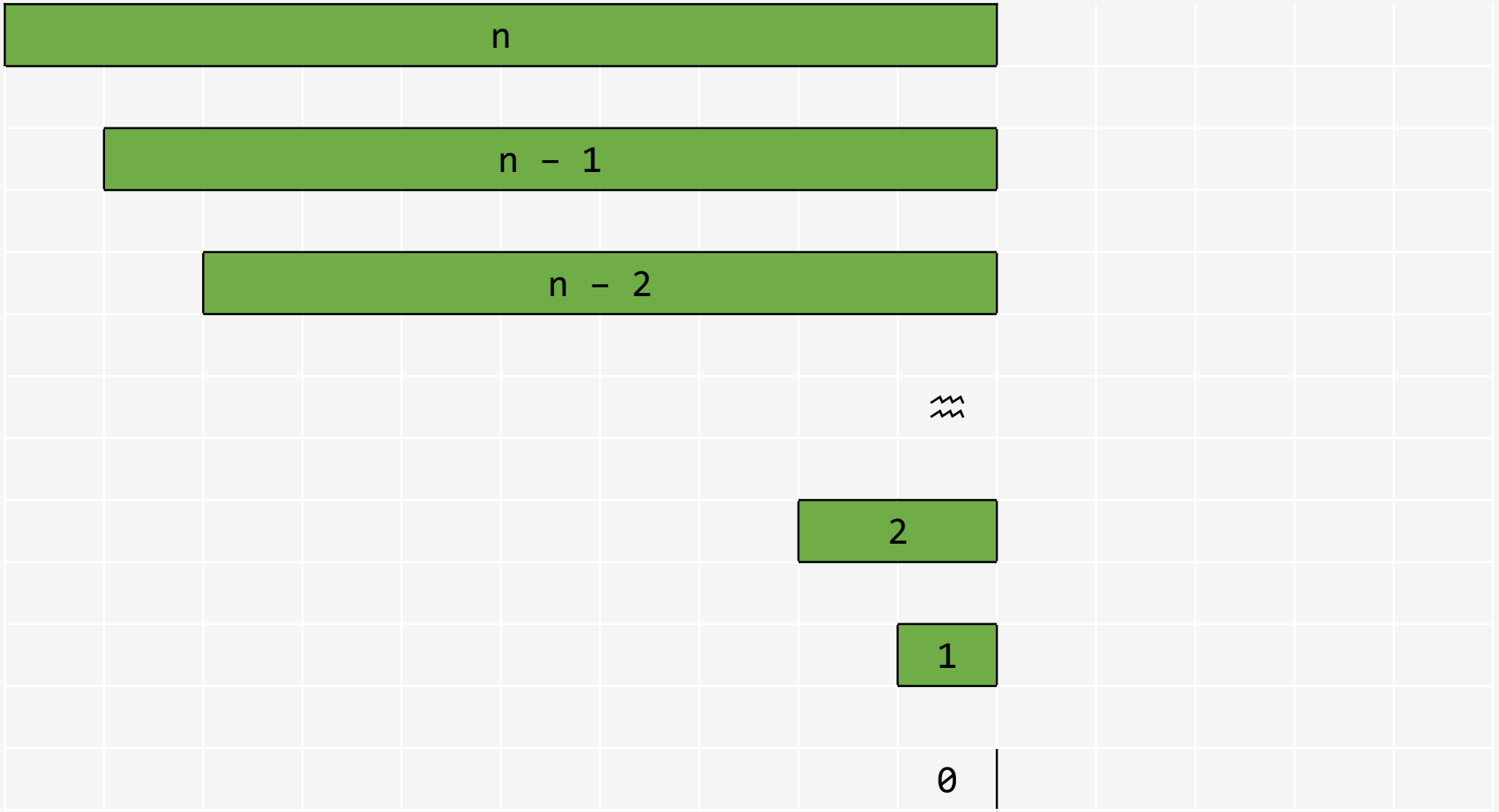
```
LinearSearch(A, low, high, key)
      if high < low:
         return NOT_FOUND
      if A[low] = key:
         return low
      return LinearSearch(A, low + 1, high, key)
```

Recurrence defining worst-case time:

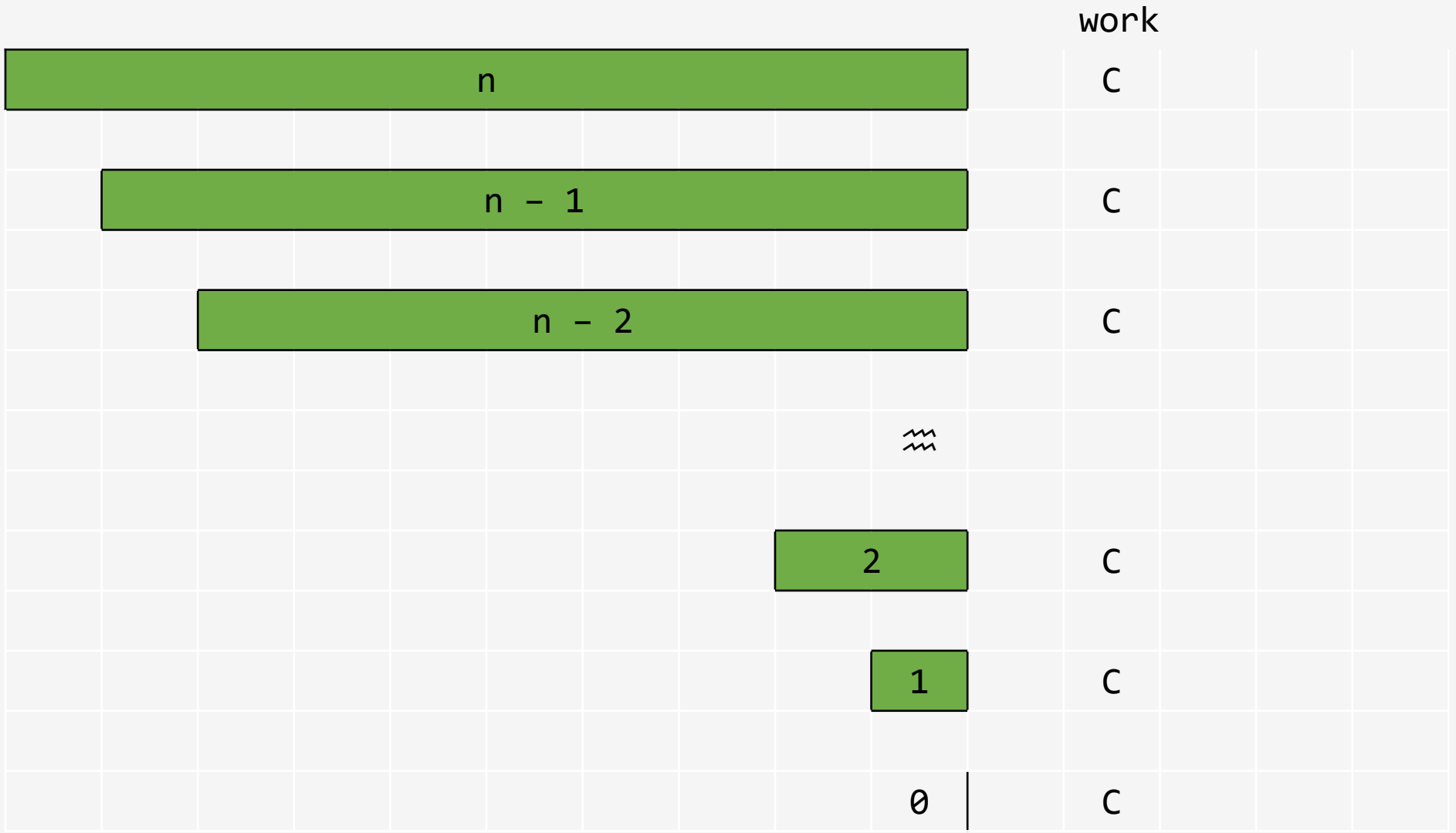
$$T(n) = T(n - 1) + c$$

$$T(0) = c$$

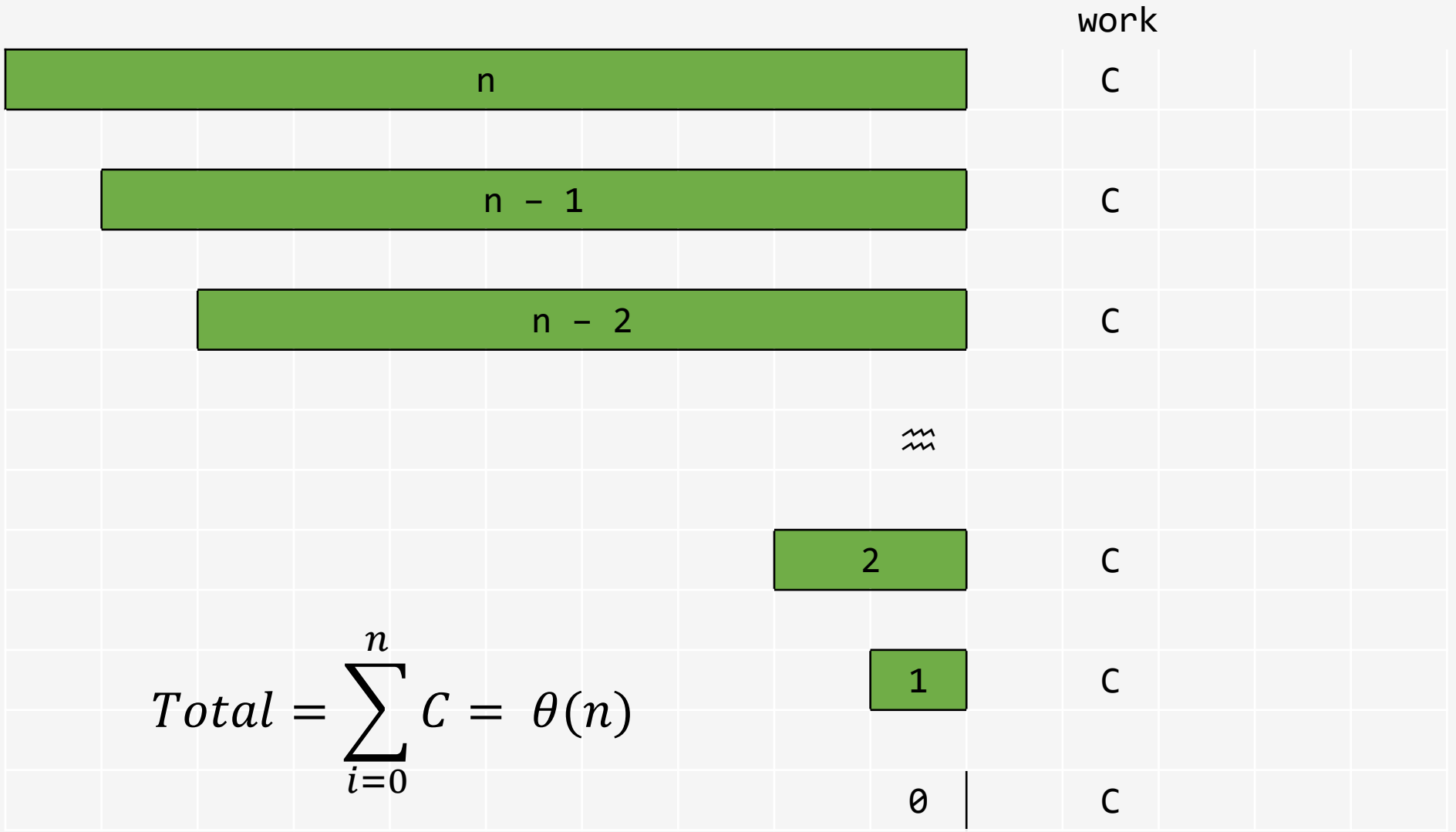
Runtime of Linear Search



Runtime of Linear Search



Runtime of Linear Search



$$Total = \sum_{i=0}^{n} C = \theta(n)$$

Iterative Version

```
LinearSearchIt(A, low, high, key)
      for i from low to high:
         if A[i] = key:
            return i
      return NOT_FOUND
```

Summary

Create a recursive solution

Define a corresponding recurrence relation, `T`

Determine `T(n)`: worst-case runtime

Optionally, create iterative solution