

## Problema di selezione di attività

$$A = \{e_1, e_2, \dots, e_n\} \text{ ATTIVITÀ}$$

usano una risorsa mutuamente esclusiva

$e_i$  ha  $s_i$  e  $f_i$  ovvero inizio e fine

Due attività sono compatibili se:

$$f_i \leq s_j \quad \text{o} \quad f_j \leq s_i$$

Se sono compatibili possono assegnare a entrambi le risorse

Dobbiamo trovare un sottoinsieme di attività compatibili che abbia il massimo delle attività possibili

DEFINIAMO IL PROBLEMA RICORSIVAMENTE

$S_{i,j} \rightarrow C[i,j]$  dimensione di una soluzione

$$C[i,j] = \begin{cases} 0 & \text{se } S_{i,j} = \emptyset \\ \max \{ C(i,k) + C(k,j) + 1 \} \end{cases}$$

↳ Dimensione totale della soluzione

Il problema definisce una sottostruttura ottimale

Si potrebbe risolvere il problema con la programmazione dinamica con memorizzazione, ma esiste un metodo migliore per la risoluzione

metodo migliore per la risoluzione

Andiamo a scegliere l'attività che termine dopo tutte

Quest'approccio viene chiamato scelta greedy

## ALGORITHMO RICORSIVO

ACTIVITY\_SELECTOR ( $F, S, m, i$ )

$m \leftarrow i+1$

while ( $m \leq n$  AND  $S[m] < F[i]$ )

$m \leftarrow m+1$

IF ( $m \leq n$ ) THEN

RETURN  $\{a_m\}$  AND ACTIVITY\_SELECTOR ( $F, S, m, m$ )

ELSE RETURN  $\emptyset$

## ALGORITHMO ITERATIVO

ACTIVITY\_SELECTOR ( $F, S, n$ )

$B \leftarrow \{a_1\}$

$i \leftarrow 1$

FOR  $m \leftarrow 2$  TO  $n$  DO

IF ( $S[m] \geq F[i]$ ) THEN

$B \leftarrow \{a_m\}$

$i \leftarrow m$

RETURN  $B$