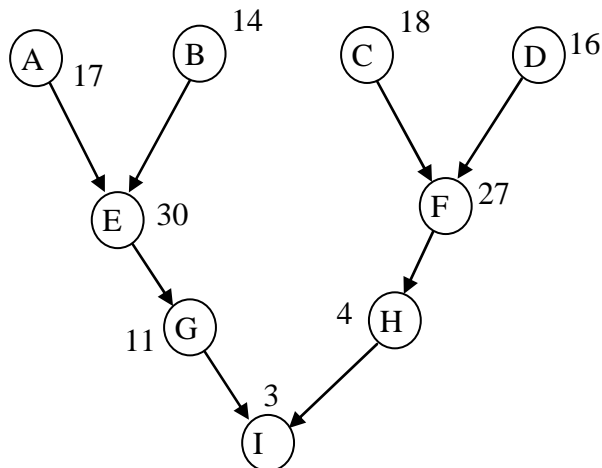


# ASSEMBLY LINE BALANCING

## Algoritmo OptPack

### DESCRIZIONE DEL PROBLEMA

E' dato il grafo di assemblaggio in figura. Si desidera progettare una linea di assemblaggio con tempo ciclo  $T_C = 40$  e minimo numero di stazioni utilizzando l'algoritmo OptPack partendo da un upper bound iniziale pari a 6. Evidenziare le soluzioni parziali costruite dall'algoritmo, la struttura dati e la sua evoluzione durante l'esecuzione.



### SOLUZIONE

1. Calcolo del massimo tempo residuo. Poiché  $UB=6$  abbiamo 240 unità di tempo a disposizione, delle quali  $17+14+18+16+30+27+11+4+3=140$  occupate dal processamento dei task e 100 disponibili per il residuo. Pertanto il max residuo  $R=100$ .
2. Un lower bound dell'ottimo è il valore  $LB_1$  di FABLE:

$$LB_1 = \left\lceil \frac{\sum_i p_i}{T_C} \right\rceil = \left\lceil \frac{17+14+18+16+30+27+11+4+3}{40} \right\rceil = 4$$

Cerchiamo pertanto soluzioni da 4,5,6 macchine.

#### Procedura di enumerazione

1. AB | CD | E | FG | HI – Trovata nuova soluzione. Aggiorniamo  $UB=4$  e  $R=4*40-140=20$ .
2. AB | CD | FH – *Not Enough Time* ( $R=20-9-6-9 < 0$ )
3. AB | D – *Station dominance*
4. AB | E | CD – *Not Going Forward* (equivale alla *Labeling dominance* di FABLE)
5. AB | E | CG – *Not Enough Time*
6. AB | E | DG – *Not Enough Time*
7. AB | E | G – *station dominance*
8. AC | BD – *Not Going Forward*
9. AC | D – *station dominance*
10. AD | BC – *Not Going Forward*
11. AD | C – *station dominance*
12. BC | AD – *Not Going Forward*
13. BD | AC – *Not Going Forward*
14. CD | AB – *Not Going Forward*
15. CD | B – *station dominance*
16. CD | FH | AB – *Not Enough Time*
17. CD | FH | B – *station dominance*
18. D – *station dominance*

La procedura termina. La soluzione corrente di valore  $UB$  è ottima (cioè quella trovata al passo 1).

Qui sotto è riportata la struttura dati al termine della procedura

### Struttura dati

