

**Esercizi proposti nel Cap. 4 - Soluzioni**

**Esercizio 4.1**

Al fine di utilizzare l'algoritmo più efficiente, verifichiamo preliminarmente l'aciclicità del digrafo. La seguente tabella riporta il dettaglio dell'algoritmo di ordinamento topologico: all'interno di ogni cella della tabella è riportata la coppia (*grado\_interno(i); f(i)*).

	Ordinamento topologico: iterazioni						
$i \in \mathcal{N}$	1°	2°	3°	4°	5°	6°	7°
$s$	<b>0; 1</b>						
1	1; -	<b>0; 2</b>					
2	3; -	2; -	1; -				
3	3; -		2; -				
4	2; -		1; -				
5	1; -						
6	3; -		2; -				

L'algoritmo si arresta alla 3° iterazione in quanto non ci sono nodi con grado interno residuo nullo: il digrafo non è aciclico. Essendo i costi sugli archi non tutti non-negativi occorre impiegare un algoritmo a correzione di etichetta ad esempio quello di Bellman-Ford. La seguente (doppia) tabella riporta il dettaglio di tale algoritmo: all'interno di ogni cella della riga associata alla coda  $Q$  è riportato il contenuto di  $Q$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia ( $d_i; pred(i)$ ); la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

	Algoritmo di Bellman-Ford: iterazioni						
	1°	2°	3°	4°	5°	6°	7°
$Q$	( $s$ )	(1, 2)	(2, 3, 4, 6)	(3, 4, 6)	(4, 6)	(6, 5)	(5)
$s$	*						
1	5; $s$	*					
2	6; $s$		*				
3		7; 1		*			2; 5
4		10; 1		-1; 3	*		
5					3; 4		*
6		7; 1			1; 4	*	

	(segue) iterazioni						
	8°	9°	10°	11°	12°	13°	14°
$Q$	(3)	(4)	(5, 6)	(6, 2, 3)	(2, 3)	(3)	(4)
$s$							
1							
2			1; 5		*		
3	*		-3; 5			*	
4	-6; 3	*				-11; 3	*
5		-2; 4	*				-7; 4
6		-4; 4		*			-9; 4

Le sequenza di iterazioni 8, 9, 10, 11, e 12 si ripete ciclicamente dalla 13 in poi comportando una diminuzione di 5 unità delle etichette  $d_i$  dei nodi  $i = 2, 3, 4, 5$  e 6, ad ogni ciclo di iterazioni. In particolare quando l'etichetta  $d_4$  del nodo 4 assume valore  $d_4 < (n - 1) c_{\min} = (7 - 1) (-8) = -48$ , abbiamo la prova che l'algoritmo non consente di determinare la soluzione del problema dei cammini minimi da  $s$  agli altri nodi per la presenza di un ciclo di costo negativo. L'algoritmo è in grado di intercettare il ciclo di costo negativo che non consente di risolvere il problema: tracciando a ritroso i predecessori a partire dal nodo 4 si intercetta il ciclo di costo negativo  $C = (4, 5, 3, 4)$ . Ovviamente, siccome l'algoritmo non è in grado di determinare la soluzione ottima per la presenza del ciclo  $C$  di costo negativo, non ha senso fornire il certificato di ottimalità della soluzione.

### Esercizio 4.2

La seguente tabella riporta il dettaglio dell'algoritmo di Dijkstra: all'interno di ogni cella della riga associata alla lista  $L$  è riportato il contenuto di  $L$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associato alla colonna).

	Algoritmo di Dijkstra: iterazioni						
	1°	2°	3°	4°	5°	6°	7°
$L$	{1}	{2, 3, 5}	{3, 4, 5}	{4, 5, 6}	{5, 6, 7}	{6, 7}	{6}
1	*						
2	3; 1	*					
3	4; 1		*				
4		4; 2		*			
5	6; 1				*		
6			8; 3	7; 4			*
7				6; 4		*	

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	1	2	3	4	5	6	7
$d_i^*$	0	3	4	4	6	7	6
$pred^*(i)$	1	1	1	2	1	4	4

La seguente tabella riporta il dettaglio dell'algoritmo di Bellman-Ford: all'interno di ogni cella della riga associata alla coda  $Q$  è riportato il contenuto di  $Q$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo di Bellman-Ford: iterazioni							
	1°	2°	3°	4°	5°	6°	7°
$Q$	(1)	(2, 3, 5)	(3, 5, 4)	(5, 4, 6)	(4, 6, 7)	(6, 7)	(7)
1	*						
2	3; 1	*					
3	4; 1		*				
4		4; 2			*		
5	6; 1			*			
6			8; 3		7; 4	*	
7				8; 5	6; 4		*

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	1	2	3	4	5	6	7
$d_i^*$	0	3	4	4	6	7	6
$pred^*(i)$	1	1	1	2	1	4	4

La seguente tabella riporta il dettaglio dell'algoritmo di D'Esopo-Pape: all'interno di ogni cella della riga associata alla coda  $Q$  è riportato il contenuto di  $Q$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo di D'Esopo-Pape: iterazioni							
	1°	2°	3°	4°	5°	6°	7°
$Q$	(1)	(2, 3, 5)	(3, 5, 4)	(5, 4, 6)	(4, 6, 7)	(7, 6)	(6)
1	*						
2	3; 1	*					
3	4; 1		*				
4		4; 2			*		
5	6; 1			*			
6			8; 3		7; 4		*
7				8; 5	6; 4	*	

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	1	2	3	4	5	6	7
$d_i^*$	0	3	4	4	6	7	6
$pred^*(i)$	1	1	1	2	1	4	4

La seguente tabella riporta il dettaglio dell'algoritmo SLF: all'interno di ogni cella della riga associata alla coda  $Q$  è riportato il contenuto di  $Q$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo SLF: iterazioni							
	1°	2°	3°	4°	5°	6°	7°
$Q$	(1)	(2, 3, 5)	(4, 3, 5)	(3, 5, 6, 7)	(5, 6, 7)	(6, 7)	(6)
1	*						
2	3; 1	*					
3	4; 1			*			
4		4; 2	*				
5	6; 1				*		
6			7; 4			*	
7			6; 4				*

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	1	2	3	4	5	6	7
$d_i^*$	0	3	4	4	6	7	6
$pred^*(i)$	1	1	1	2	1	4	4

### Esercizio 4.3

Assumendo la data arborecenza esterna  $T = (\mathcal{N}, \mathcal{A}(T))$  radicata in  $s$  quella dei cammini minimi (albero dei cammini minimi) della rete  $R = (\mathcal{N}, \mathcal{A}, \mathbf{c})$ , si ricava il vettore  $\mathbf{d}$  delle distanze minime dei nodi dal nodo  $s$ , calcolate imponendo  $d_j = d_i + c_{ij}$ , per ogni arco  $(i, j) \in \mathcal{A}(T)$  dell'arborecenza, avendo posto  $d_s = 0$ :

$i \in \mathcal{N}$	$s$	$a$	$b$	$c$	$d$	$e$
$d_i$	0	3	4	9	7	12

Applichiamo la condizione di ottimalità di Bellman sugli archi della rete. È sufficiente applicarla sugli archi di  $\mathcal{A} \setminus \mathcal{A}(T)$ , essendo per costruzione verificate (all'uguaglianza) sugli archi dell'arborecenza:

$$(b, a): \quad d_a \leq d_b + c_{ba}; \quad 3 \leq 4 + 3, \quad \text{SÌ}$$

$$\begin{array}{lll}
(b, c): & d_c \leq d_b + c_{bc}; & 9 \leq 4 + 4, \quad \text{NO} \\
(d, c): & d_c \leq d_d + c_{dc}; & 9 \leq 7 + 2, \quad \text{SÌ} \\
(d, e): & d_e \leq d_d + c_{de}; & 12 \leq 7 + 5, \quad \text{SÌ}
\end{array}$$

La condizione non è soddisfatta sull'arco  $(b, c)$  quindi l'arborescenza data non rappresenta l'albero dei communi minimi dal nodo  $s$ .

### Esercizio 4.4

La formulazione del problema (P) come problema di flusso su rete a costo minimo sulla rete  $R = (\mathcal{N}, \mathcal{A}, \mathbf{b}, \mathbf{c}, \mathbf{l}, \mathbf{u})$ , con  $l_{ij} = 0$  e  $u_{ij} = +\infty$ ,  $(i, j) \in \mathcal{A}$ , e ponendo  $b_s = 1$ ,  $b_t = -1$  e  $b_i = 0$ ,  $i \in \mathcal{N} \setminus \{s, t\}$ , è la seguente:

$$\min z(\mathbf{x}) = \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}$$

s. v.

$$\begin{aligned}
\sum_{j \in \mathcal{N}: (s,j) \in \mathcal{A}} x_{sj} - \sum_{h \in \mathcal{N}: (h,s) \in \mathcal{A}} x_{hs} &= 1 \\
\sum_{j \in \mathcal{N}: (i,j) \in \mathcal{A}} x_{ij} - \sum_{h \in \mathcal{N}: (h,i) \in \mathcal{A}} x_{hi} &= 0, \quad i \in \mathcal{N} \setminus \{s, t\}, \\
\sum_{j \in \mathcal{N}: (t,j) \in \mathcal{A}} x_{tj} - \sum_{h \in \mathcal{N}: (h,t) \in \mathcal{A}} x_{ht} &= -1, \\
x_{ij} &\geq 0, \quad (i, j) \in \mathcal{A},
\end{aligned}$$

cioè:

$$\min z(\mathbf{x}) = 4x_{s1} + 2x_{s2} + 2x_{13} + 2x_{24} + x_{25} - 5x_{34} + 2x_{3t} + x_{41} + 5x_{4t} + 2x_{54} + 2x_{5t}$$

s. v.

$$\begin{aligned}
x_{s1} + x_{s2} &= 1 \\
x_{13} - x_{s1} - x_{41} &= 0 \\
x_{24} + x_{25} - x_{s2} &= 0 \\
x_{34} + x_{3t} - x_{13} &= 0 \\
x_{41} + x_{4t} - x_{24} - x_{34} - x_{54} &= 0 \\
x_{54} + x_{5t} - x_{25} &= 0 \\
-x_{3t} - x_{4t} - x_{5t} &= -1 \\
x_{s1}, x_{s2}, x_{13}, x_{24}, x_{25}, x_{34}, x_{3t}, x_{41}, x_{4t}, x_{54}, x_{5t} &\geq 0.
\end{aligned}$$

Attraverso il risolutore di Excel si ricava che la soluzione ottima è illimitata inferiormente, per la presenza del ciclo di costo negativo  $(1, 3, 4, 1)$ .

La soluzione ottima del problema (P'), ottenuto vincolando le variabili ad assumere valori non maggiori di uno, è:

$$x_{s2}^*, x_{13}^*, x_{25}^*, x_{34}^*, x_{41}^*, x_{5t}^* = 1.$$

Questa non rappresenta un cammino (minimo), bensì il cammino  $(s, 2, 5, t)$  e il ciclo  $(1, 3, 4, 1)$  tra loro disconnessi.

### Esercizio 4.5

Al fine di utilizzare l'algoritmo più efficiente, verifichiamo preliminarmente l'aciclicità del digrafo  $D = (\mathcal{N}, \mathcal{A})$ . La seguente tabella riporta il dettaglio dell'algoritmo di ordinamento topologico: all'interno di ogni cella

della tabella è riportata la coppia  $(grado\_interno(i); f(i))$ .

	Ordinamento topologico: iterazioni						
$i \in \mathcal{N}$	1°	2°	3°	4°	5°	6°	7°
$s$	<b>0; 1</b>						
1	1; -	<b>0; 2</b>					
2	2; -	1; -	<b>0; 3</b>				
3	2; -		1; -	<b>0; 4</b>			
4	2; -		1; -		<b>0; 5</b>		
5	3; -			2; -	1; -	<b>0; 6</b>	
$t$	3; -			2; -		1; -	<b>0; 7</b>

L' algoritmo di ordinamento topologico termina assegnando numerazione a tutti i nodi: il digrafo è aciclico. E' quindi possibile applicare l' algoritmo dei cammini minimi per le reti acicliche sulla base della numerazione topologica  $f(i)$ ,  $i \in \mathcal{N}$ , individuata:

$i \in \mathcal{N}$	$s$	1	2	3	4	5	$t$
$f(i)$	1	2	3	4	5	6	7

Lo svolgimento dell' algoritmo è:

$f(i)$	$i \in \mathcal{N}$	$d^*_i$	$pred(i)$
1	$s$	$d^*_s = 0$	$s$
2	1	$d^*_1 = d^*_s + c_{s1} = 6$	$s$
3	2	$d^*_2 = \min\{d^*_s + c_{s2}; d^*_1 + c_{12}\} = 4$	$s$
4	3	$d^*_3 = \min\{d^*_1 + c_{13}; d^*_2 + c_{23}\} = 8$	1
5	4	$d^*_4 = \min\{d^*_1 + c_{14}; d^*_3 + c_{34}\} = 10$	3
6	5	$d^*_5 = \min\{d^*_2 + c_{25}; d^*_3 + c_{35}; d^*_4 + c_{45}\} = 6$	2
7	$t$	$d^*_t = \min\{d^*_2 + c_{2t}; d^*_4 + c_{4t}; d^*_5 + c_{5t}\} = 11$	2

L' arborescenza esterna  $T^* = (\mathcal{N}, \mathcal{A}(T^*))$  radicata in  $s$  rappresenta l' albero dei cammini minimi da  $s$ , con  $\mathcal{A}(T^*) = \{(pred(1) = s, 1), (pred(2) = s, 2), (pred(3) = 1, 3), (pred(5) = 2, 5), (pred(t) = 2, t), (pred(4) = 3, 4)\}$ .

Per giustificare l' ottimalità della soluzione individuata, applichiamo la condizione di ottimalità di Bellman sugli archi del digrafo. E' sufficiente applicarla sugli archi di  $\mathcal{A} \setminus \mathcal{A}(T^*)$ , essendo per costruzione verificate (all' uguaglianza) sugli archi dell' arborescenza:

$$\begin{aligned}
 (1, 2): \quad & d^*_2 \leq d^*_1 + c_{12}; \quad 4 \leq 6 + 3, \quad \text{SÌ} \\
 (1, 4): \quad & d^*_4 \leq d^*_1 + c_{14}; \quad 10 \leq 6 + 7, \quad \text{SÌ} \\
 (2, 3): \quad & d^*_3 \leq d^*_2 + c_{23}; \quad 8 \leq 4 + 8, \quad \text{SÌ} \\
 (3, 5): \quad & d^*_5 \leq d^*_3 + c_{35}; \quad 6 \leq 8 + 2, \quad \text{SÌ} \\
 (4, 5): \quad & d^*_5 \leq d^*_4 + c_{45}; \quad 6 \leq 10 + 5, \quad \text{SÌ} \\
 (4, t): \quad & d^*_t \leq d^*_4 + c_{4t}; \quad 11 \leq 10 + 1, \quad \text{SÌ} \\
 (5, t): \quad & d^*_t \leq d^*_5 + c_{5t}; \quad 11 \leq 6 + 7, \quad \text{SÌ}
 \end{aligned}$$

Essendo la condizione di ottimalità di Bellman soddisfatta su tutti gli archi, la soluzione individuata è ottima.

### Esercizio 4.6

La seguente tabella riporta il dettaglio dell'algoritmo di Dijkstra: all'interno di ogni cella della riga associata alla lista  $L$  è riportato il contenuto di  $L$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo di Dijkstra: iterazioni								
	1°	2°	3°	4°	5°	6°	7°	8°
$L$	{s}	{1, 4, 7}	{2,3,4,7}	{2, 4, 7}	{4, 5, 7}	{5, 6, 7}	{6, 7}	{6}
$s$	*							
1	4; s	*						
2		10; 1	6; 3	*				
3		5; 1	*					
4	7; s		6; 3		*			
5				8; 2		*		
6					9; 4			*
7	8; s						*	

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	$s$	1	2	3	4	5	6	7
$d_i^*$	0	4	6	5	6	8	9	8
$pred^*(i)$	$s$	$s$	3	1	3	2	4	$s$

### Esercizio 4.7

La formulazione del problema (P) del cammino minimo sulla rete  $R = (\mathcal{N}, \mathcal{A}, \mathbf{c})$  dal nodo 1 al nodo 5 è la seguente:

$$\min z(\mathbf{x}) = \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}$$

s. v.

$$\begin{aligned} \sum_{j \in \mathcal{N}: (1,j) \in \mathcal{A}} x_{1j} - \sum_{h \in \mathcal{N}: (h,1) \in \mathcal{A}} x_{h1} &= 1 \\ \sum_{j \in \mathcal{N}: (i,j) \in \mathcal{A}} x_{ij} - \sum_{h \in \mathcal{N}: (h,i) \in \mathcal{A}} x_{hi} &= 0, \quad i \in \mathcal{N} \setminus \{1, 5\}, \\ \sum_{j \in \mathcal{N}: (5,j) \in \mathcal{A}} x_{5j} - \sum_{h \in \mathcal{N}: (h,5) \in \mathcal{A}} x_{h5} &= -1, \\ x_{ij} &\geq 0, \quad (i,j) \in \mathcal{A} \end{aligned}$$

cioè:

$$\min z(\mathbf{x}) = \alpha^2 x_{12} + 4\alpha x_{23} + \alpha^2 x_{24} + 4x_{31} + 4\alpha x_{34} - 4x_{45} - 8x_{53}$$

s. v.

$$\begin{aligned}
x_{12} - x_{31} &= 1 \\
x_{23} + x_{24} - x_{12} &= 0 \\
x_{31} + x_{34} - x_{23} - x_{53} &= 0 \\
x_{45} - x_{24} - x_{34} &= 0 \\
x_{53} - x_{45} &= -1 \\
x_{12}, x_{23}, x_{24}, x_{31}, x_{34}, x_{45}, x_{53} &\geq 0.
\end{aligned}$$

La soluzione ottima è finita se i cicli della rete sono tutti di costo non negativo:

$$\begin{aligned}
(1, 2, 3, 1): \quad & c_{12} + c_{23} + c_{31} \geq 0; & \alpha^2 + 4\alpha + 4 \geq 0; & \alpha \neq -2 \\
(1, 2, 4, 5, 3, 1): \quad & c_{12} + c_{24} + c_{45} + c_{53} + c_{31} \geq 0; & \alpha^2 + \alpha^2 - 4 - 8 + 4 \geq 0; & \alpha \leq -2, \alpha \geq 2 \\
(3, 4, 5, 3): \quad & c_{34} + c_{45} + c_{53} \geq 0; & 4\alpha - 4 - 8 \geq 0; & \alpha \geq 3.
\end{aligned}$$

Pertanto,  $I_\alpha = [3, +\infty)$ .

Posto  $\alpha = \min\{\alpha : \alpha \geq 3, \alpha > 0\} = 3$ , la seguente tabella riporta il dettaglio dell'algoritmo SLF: all'interno di ogni cella della riga associata alla coda  $Q$  è riportato il contenuto di  $Q$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo SLF: iterazioni					
	1°	2°	3°	4°	5°
$Q$	(1)	(2)	(4, 3)	(5, 3)	(3)
1	*				
2	9; 1	*			
3		21; 2		6; 5	*
4		18; 2	*		
5			14; 4	*	

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo 1 e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo 1 sono:

$i \in \mathcal{N}$	1	2	3	4	5
$d_i^*$	0	9	6	18	14
$pred^*(i)$	1	1	5	2	4

### Esercizio 4.8

Le matrici nelle seguenti tabelle riportano i risultati delle iterazioni dell'algoritmo. In grassetto sono riportati i valori aggiornati nelle varie iterazioni.



0	7	2	4
$+\infty$	0	6	$+\infty$
$+\infty$	$+\infty$	0	-3
$+\infty$	4	$+\infty$	0

$$d_{[i,j]}^{(0)}, i = 1, \dots, 4, j = 1, \dots, 4$$

1	1	1	1
1	2	2	4
1	2	3	3
1	4	3	4

$$pred_{[i,j]}^{(0)}, i = 1, \dots, 4, j = 1, \dots, 4$$

0	7	2	4
$+\infty$	0	6	$+\infty$
$+\infty$	$+\infty$	0	-3
$+\infty$	4	$+\infty$	0

$$d_{[i,j]}^{(1)}, i = 1, \dots, 4, j = 1, \dots, 4$$

1	1	1	1
1	2	2	4
1	2	3	3
1	4	3	4

$$pred_{[i,j]}^{(1)}, i = 1, \dots, 4, j = 1, \dots, 4$$

0	7	2	4
$+\infty$	0	6	$+\infty$
$+\infty$	$+\infty$	0	-3
$+\infty$	4	<b>10</b>	0

$$d_{[i,j]}^{(2)}, i = 1, \dots, 4, j = 1, \dots, 4$$

1	1	1	1
1	2	2	4
1	2	3	3
1	4	<b>2</b>	4

$$pred_{[i,j]}^{(2)}, i = 1, \dots, 4, j = 1, \dots, 4$$

0	7	2	<b>-1</b>
$+\infty$	0	6	<b>3</b>
$+\infty$	$+\infty$	0	-3
$+\infty$	4	10	0

$$d_{[i,j]}^{(3)}, i = 1, \dots, 4, j = 1, \dots, 4$$

1	1	1	<b>3</b>
1	2	2	<b>3</b>
1	2	3	3
1	4	2	4

$$pred_{[i,j]}^{(3)}, i = 1, \dots, 4, j = 1, \dots, 4$$

0	<b>3</b>	2	-1
$+\infty$	0	6	3
$+\infty$	<b>1</b>	0	-3
$+\infty$	4	10	0

$$d_{[i,j]}^{(4)}, i = 1, \dots, 4, j = 1, \dots, 4$$

1	<b>4</b>	1	3
1	2	2	3
1	<b>4</b>	3	3
1	4	2	4

$$pred_{[i,j]}^{(4)}, i = 1, \dots, 4, j = 1, \dots, 4$$

Le etichette  $d_{[i,j]}^{(4)}$  e i nodi  $pred_{[i,j]}^{(4)}$  rappresentano, rispettivamente, il costo  $d_{[i,j]}^*$  del percorso orientato minimo da  $i$  a  $j$  e il nodo  $pred_{[i,j]}^*$ , predecessore del nodo  $j$  in tale percorso, per ogni  $[i,j] \in \mathcal{N} \times \mathcal{N}$ .

### Esercizio 4.9

La formulazione del problema (P) del cammino minimo sulla rete  $R = (\mathcal{N}, \mathcal{A}, \mathbf{c})$  dal nodo CP a tutti gli altri nodi  $CL_i \in \mathcal{N} \setminus \{CP\}$  è la seguente:

$$\min z(\mathbf{x}) = \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}$$

s. v.

$$\sum_{j \in \mathcal{N}: (CP, j) \in \mathcal{A}} x_{CP, j} - \sum_{h \in \mathcal{N}: (h, CP) \in \mathcal{A}} x_{h, CP} = 6$$

$$\sum_{j \in \mathcal{N}: (CL_i, j) \in \mathcal{A}} x_{CL_i, j} - \sum_{h \in \mathcal{N}: (h, CL_i) \in \mathcal{A}} x_{h, CL_i} = -1, CL_i \in \mathcal{N} \setminus \{CP\},$$

$$x_{ij} \geq 0, (i, j) \in \mathcal{A}$$

cioè:

$$\min z(\mathbf{x}) = 20x_{CP,CL_1} + 40x_{CP,CL_2} + 60x_{CP,CL_3} + 60x_{CP,CL_4} + 10x_{CL_1,CL_2} + 30x_{CL_2,CL_3} + 10x_{CL_2,CL_5} + 20x_{CL_3,CL_5} + 10x_{CL_4,CL_1} + 20x_{CL_4,CL_2} + \alpha x_{CL_4,CL_6} + 10x_{CL_5,CL_4} + 30x_{CL_5,CL_6}$$

s. v.

$$\begin{aligned} x_{CP,CL_1} + x_{CP,CL_2} + x_{CP,CL_3} + x_{CP,CL_4} &= 6 \\ x_{CL_1,CL_2} - x_{CP,CL_1} - x_{CL_4,CL_1} &= -1 \\ x_{CL_2,CL_3} + x_{CL_2,CL_5} - x_{CP,CL_2} - x_{CL_1,CL_2} - x_{CL_4,CL_2} &= -1 \\ x_{CL_3,CL_5} - x_{CP,CL_3} - x_{CL_2,CL_3} &= -1 \\ x_{CL_4,CL_1} + x_{CL_4,CL_2} + x_{CL_4,CL_6} - x_{CP,CL_4} - x_{CL_5,CL_4} &= -1 \\ x_{CL_5,CL_4} + x_{CL_5,CL_6} - x_{CL_2,CL_5} - x_{CL_3,CL_5} &= -1 \\ -x_{CL_4,CL_6} - x_{CL_5,CL_6} &= -1 \\ x_{CP,CL_1}, x_{CP,CL_2}, x_{CP,CL_3}, x_{CP,CL_4}, x_{CL_1,CL_2}, x_{CL_2,CL_3} &\geq 0 \\ x_{CL_2,CL_5}, x_{CL_3,CL_5}, x_{CL_4,CL_1}, x_{CL_4,CL_2}, x_{CL_4,CL_6}, x_{CL_5,CL_4}, x_{CL_5,CL_6} &\geq 0. \end{aligned}$$

Posto  $c_{CL_4,CL_6} = \alpha = 3$ , la seguente tabella riporta il dettaglio dell'algoritmo di Dijkstra: all'interno di ogni cella della riga associata alla lista  $L$  è riportato il contenuto di  $L$  all'inizio dell'iterazione; all'interno di ogni cella della riga associata al nodo  $i$  è riportata la coppia  $(d_i; pred(i))$ ; la cella asteriscata indica il nodo (associato alla riga) selezionato nell'iterazione (associata alla colonna).

Algoritmo di Dijkstra: iterazioni							
	1°	2°	3°	4°	5°	6°	7°
$L$	{CP}	{CL <sub>1</sub> , CL <sub>2</sub> , CL <sub>3</sub> , CL <sub>4</sub> }	{CL <sub>2</sub> , CL <sub>3</sub> , CL <sub>4</sub> }	{CL <sub>3</sub> , CL <sub>4</sub> , CL <sub>5</sub> }	{CL <sub>3</sub> , CL <sub>4</sub> , CL <sub>6</sub> }	{CL <sub>3</sub> , CL <sub>6</sub> }	{CL <sub>3</sub> }
CP	*						
CL <sub>1</sub>	20; CP	*					
CL <sub>2</sub>	40; CP	30; CL <sub>1</sub>	*				
CL <sub>3</sub>	60; CP						*
CL <sub>4</sub>	60; CP			50; CL <sub>5</sub>	*		
CL <sub>5</sub>			40; CL <sub>2</sub>	*			
CL <sub>6</sub>				70; CL <sub>5</sub>	53; CL <sub>4</sub>	*	

Il vettore  $\mathbf{d}^*$  delle distanze minime dei nodi dal nodo CP e il vettore  $\mathbf{pred}^*$  dei predecessori nell'albero dei cammini minimi dal nodo CP sono:

$i \in \mathcal{N}$	CP	CL <sub>1</sub>	CL <sub>2</sub>	CL <sub>3</sub>	CL <sub>4</sub>	CL <sub>5</sub>	CL <sub>6</sub>
$d_i^*$	0	20	30	60	50	40	53
$pred^*(i)$	CP	CP	CL <sub>1</sub>	CP	CL <sub>5</sub>	CL <sub>2</sub>	CL <sub>4</sub>

da cui si ricava l'arborescenza esterna  $T^* = (\mathcal{N}, \mathcal{A}(T^*))$  radicata in CP dei cammini orientati minimi, dove  $\mathcal{A}(T^*) = \{(pred^*(CL_1) = CP, CL_1), (pred^*(CL_3) = CP, CL_3), (pred^*(CL_2) = CL_1, CL_2), (pred^*(CL_5) = CL_2, CL_5), (pred^*(CL_6) = CL_4, CL_6), (pred^*(CL_4) = CL_5, CL_4)\}$ .

Indicata con  $T^*(\alpha) = (\mathcal{N}, \mathcal{A}(T^*(\alpha)))$  l'arborescenza esterna dei cammini minimi da CP e indicato con  $\mathbf{d}^*(\alpha)$  il vettore delle distanze minime dei nodi, in funzione di  $c_{CL_4,CL_6} = \alpha$ , affinché l'arco  $(CL_4, CL_6)$  appartenga alla soluzione ottima, occorre che  $(CL_4, CL_6) \in \mathcal{A}(T^*(\alpha))$ . Questo in particolare richiede che:



Algoritmo di Dijkstra «a due liste»: iterazioni tabella $b$								
	1°	2°	3°	4°	5°	6°	7°	8°
$L^{(b)}$	$\{b\}$	$\{4, 5, 6\}$	$\{1, 4, 6\}$	$\{1, 2, 3, 6\}$	$\{1, 2, 3\}$	$\{\}$	$\{\}$	$\{\}$
$a$								
1		$5 + k; 5$	$5 + k; 5$ ( $k \leq 4$ )					
2			10; 4					
3			12; 4	11; 6				
4	$5; b$		*					
5	$k; b$	$*(k \leq 5)$						
6	$5; b$			*				
$b$	*							

Siccome alla 5° iterazione nella tabella  $a$  e alla 3° iterazione nella tabella  $b$  le etichette  $d_4^{(a)} = 10$  e  $d_4^{(b)} = 5$  del nodo 4 risultano rispettivamente permanenti, l'algoritmo termina all'inizio della 5° iterazione nella tabella  $a$ .

Al termine dell'algoritmo, il vettore  $\mathbf{d}^{(a)}$  delle distanze (stime per eccesso delle minime) dei nodi del digrafo dal nodo  $a$  e il vettore **pred** dei predecessori dei nodi nei migliori cammini dal nodo  $a$  individuati sono:

$i \in \mathcal{N}$	<u><math>a</math></u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	5	6	$b$
$d_i^{(a)}$	0	6	9	7	10	11	13	$+\infty$
$pred(i)$	$a$	$a$	1	$a$	1	1	3	$b$

I nodi sottolineati sono quelli etichettati in modo permanente (e le relative etichette e predecessori sono ottimi); in particolare, indicato con  $S^{(a)}$  l'insieme dei nodi etichettati permanentemente, si ha  $S^{(a)} = \{a, 1, 2, 3, 4\}$ .

Al termine dell'algoritmo il vettore  $\mathbf{d}'^{(b)}$  delle distanze (stime per eccesso delle minime) del nodo  $b$  dai nodi del digrafo, e il vettore **succ** dei successori dei nodi nei migliori cammini verso il nodo  $b$  individuati sono:

$i \in \mathcal{N}$	$a$	1	2	3	<u>4</u>	<u>5</u>	<u>6</u>	<u><math>b</math></u>
$d'_i^{(b)}$	$+\infty$	$5 + k$ ( $k \leq 4$ )	10	11	5	$k$ ( $k \leq 5$ )	5	$+\infty$
$succ(i)$	$a$	5	4	6	$b$	$b$	$b$	$b$

I nodi sottolineati sono quelli etichettati in modo permanente (e le relative etichette e predecessori sono ottimi); in particolare, indicato con  $S^{(b)}$  l'insieme dei nodi etichettati permanentemente, si ha  $S^{(b)} = \{4, 5, 6, b\}$ .

Essendo  $S^{(a)} = \{a, 1, 2, 3, 4\}$  e  $S^{(b)} = \{4, 5, 6, b\}$  si confrontano i costi dei seguenti cammini:

$$P_{a4}^* \oplus P_{4b}^*, \text{ di costo } d_4^{*(a)} + d_4'^{*(b)} = 15;$$

$$P_{a1}^* \oplus (1, 4) \oplus P_{4b}^*, \text{ di costo } d_1^{*(a)} + c_{14} + d_4'^{*(b)} = 15;$$



Siccome alla 3° iterazione nella prima tabella e alla 2° iterazione nella seconda tabella le etichette  $d_3^{(s)} = 5$  e  $d_3^{(2)} = 1$  del nodo 3 risultano rispettivamente permanenti, l'algoritmo termina all'inizio della 3° iterazione nella prima tabella.

Al termine dell'algoritmo, il vettore  $\mathbf{d}^{(s)}$  delle distanze (stime per eccesso delle minime) dei nodi del digrafo dal nodo  $s$  e il vettore **pred** dei predecessori dei nodi nei migliori cammini dal nodo  $s$  individuati sono:

$i \in \mathcal{N}$	<u><math>s</math></u>	<u>1</u>	2	<u>3</u>	4	5	6	7
$d_i^{(a)}$	0	4	10	5	7	$+\infty$	$+\infty$	8
$pred(i)$	$s$	$s$	1	1	$s$	5	6	$s$

I nodi sottolineati sono quelli etichettati in modo permanente (e le relative etichette e predecessori sono ottimi); in particolare, indicato con  $S^{(s)}$  l'insieme dei nodi etichettati permanentemente, si ha  $S^{(s)} = \{s, 1, 3\}$ .

Al termine dell'algoritmo il vettore  $\mathbf{d}^{(2)}$  delle distanze (stime per eccesso delle minime) del nodo  $b$  dai nodi del digrafo, e il vettore **succ** dei successori dei nodi nei migliori cammini verso il nodo  $b$  individuati sono:

$i \in \mathcal{N}$	$s$	1	<u>2</u>	<u>3</u>	4	5	6	7
$d_i^{(b)}$	$+\infty$	2	0	1	2	2	$+\infty$	$+\infty$
$succ(i)$	$s$	3	2	2	3	2	6	7

I nodi sottolineati sono quelli etichettati in modo permanente (e le relative etichette e predecessori sono ottimi); in particolare, indicato con  $S^{(2)}$  l'insieme dei nodi etichettati permanentemente, si ha  $S^{(2)} = \{2, 3\}$ .

Essendo  $S^{(s)} = \{s, 1, 3\}$  e  $S^{(2)} = \{2, 3\}$  si confrontano i costi dei seguenti cammini:

$$P_{s3}^* \oplus P_{32}^*, \text{ di costo } d_3^{*(s)} + d_3'^{*(2)} = 6;$$

$$P_{s1}^* \oplus (1, 2) \oplus P_{22}^*, \text{ di costo } d_1^{*(s)} + c_{12} + d_2'^{*(2)} = 10.$$

Il cammino minimo da  $s$  a 2 è  $P_{s3}^* \oplus P_{32}^* = (s, 1, 3, 2)$  di costo 6.

Applichiamo ora l'algoritmo  $A^*$ , con il dato vettore  $h = [5 \ 1 \ 0 \ 1 \ 1 \ 1 \ 4 \ 5]^T$  delle stime per difetto (ammissibile) delle etichette ottime  $d_i'^{*(2)}$ , con  $i = s, 1, 2, 3, 4, 5, 6, 7$ . Si noti che la stima non è consistente perché  $h_7 > c_{74} + h_4$ , pertanto l'algoritmo potrebbe non comportarsi come un algoritmo ad assegnazione di etichetta e quindi non è sufficiente selezionare il nodo  $t = 2$  per determinare il cammino minimo ma l'algoritmo va iterato fin quando la lista  $L$  è vuota.

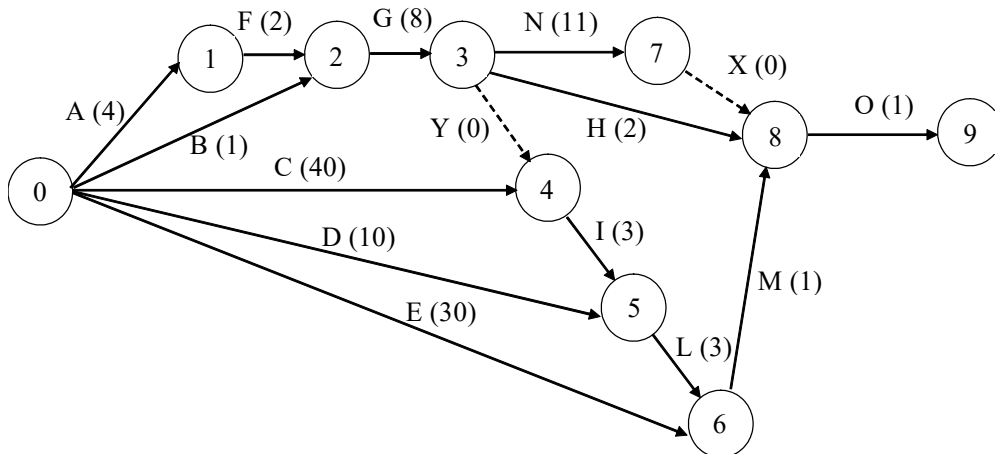
La seguente tabella riporta il *run* dell'algoritmo  $A^*$ , dove la generica cella della tabella della riga corrispondente al nodo  $j$  riporta la somma  $d_j + h_j$  seguita dal corrente  $pred(j)$ .

Algoritmo A*: iterazioni								
	1°	2°	3°	4°	5°	6°	7°	8°
$L$	$\{s\}$	$\{1, 4, 7\}$	$\{2,3,4,7\}$	$\{2, 4, 7\}$	$\{4, 5, 7\}$	$\{5, 6, 7\}$	$\{6\}$	$\{\}$
$s$	*							
1	$4 + 1; s$	*						
2		$10 + 0; 1$	$6 + 0; 3$	*				
3		$5 + 1; 1$	*					
4	$7 + 1; s$		$6 + 1; 3$		*			
5				$8 + 1; 2$		*		
6					$9 + 4; 4$			*
7	$8 + 5; s$						*	

Il cammino minimo da  $s$  a 2 è  $(s, 1, 3, 2)$  di costo 6.

### Esercizio 4.12

La rete di progetto  $R = (\mathcal{N}, \mathcal{A}, \mathbf{c})$  con attività sugli archi è rappresentata nella seguente figura. I valori tra parentesi rappresentano le durate  $c_{ij}$  delle attività. In particolare, il nodo origine 0 rappresenta l'inizio del progetto e il nodo pozzo 9 la sua fine.



La rete  $R$  è aciclica ed in particolare un suo possibile ordinamento topologico coincide con la stessa numerazione dei nodi.

Attraverso la procedura in avanti che esamina i nodi in ordine topologico, si determinano le etichette  $t_i = \max_{(h,i) \in \Gamma^-(i)} \{t_h + c_{hi}\}$ , per ogni nodo  $i$  e avendo posto  $t_0 = 0$ . Le etichette  $t_i$  così calcolate rappresentano le lunghezze (durate) dei cammini di lunghezza massima dal nodo sorgente 0 ai nodi  $i$  della rete. Per ciascun nodo  $i \in \mathcal{N}$ , la seguente tabella riporta l'etichetta  $t_i$  e il nodo predecessore  $pred(i)$  nell'albero dei cammini massimi dal nodo sorgente 0.

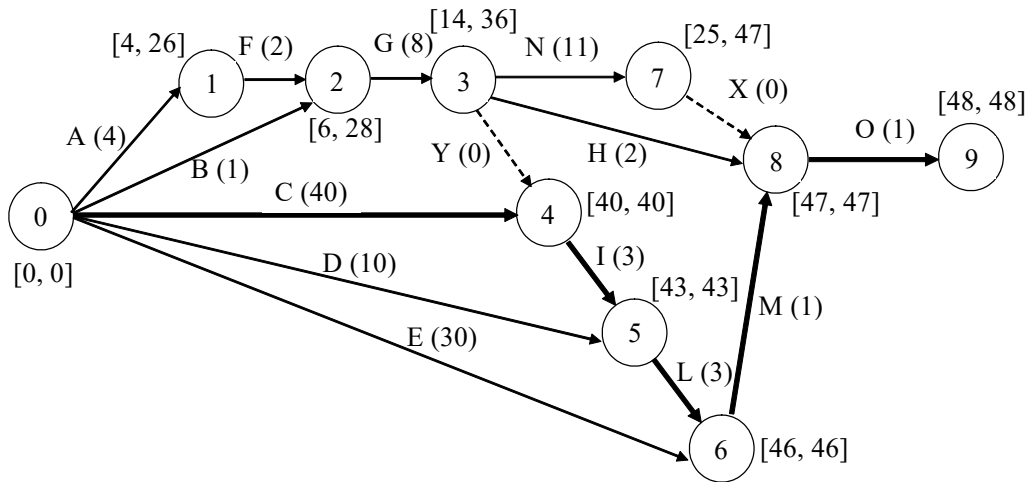
$i \in \mathcal{N}$	0	1	2	3	4	5	6	7	8	9
$t_i$	0	4	6	14	40	43	46	25	47	48
$pred(i)$	0	0	1	2	0	4	5	3	6	8

Le etichette  $t_i$  così calcolate rappresentano gli istanti minimi di occorrenza degli eventi del progetto associati ai nodi  $i \in \mathcal{N}$  della rete. In particolare, l'etichetta  $t_9$  del nodo pozzo 9 determina la durata (minima)  $T$  del progetto ( $T = t_9$ ).

Attraverso la procedura all'indietro che esamina i nodi in ordine topologico inverso, si determinano le etichette  $\bar{t}_i = \max_{(i,j) \in \Gamma^+(i)} \{\bar{t}_j + c_{ij}\}$ , per ogni nodo  $i$  e avendo posto  $\bar{t}_9 = T = t_9$ . Le etichette  $\bar{t}_i$  così calcolate rappresentano le differenze tra la durata  $T$  del progetto e le lunghezze (durate) dei cammini di lunghezza massima dai nodi  $i$  al nodo pozzo 9. Per ciascun nodo  $i \in \mathcal{N}$ , la seguente tabella riporta l'etichetta  $\bar{t}_i$  e il nodo successore  $succ(i)$  nell'albero dei cammini massimi dal nodo  $i$  al nodo pozzo 9.

$i \in \mathcal{N}$	0	1	2	3	4	5	6	7	8	9
$\bar{t}_i$	0	4	6	14	40	43	46	47	47	48
$succ(i)$	0	0	1	2	0	6	8	8	9	9

La seguente rete riporta il risultato dell'analisi temporale del progetto, con l'indicazione dei tempi minimi e massimi  $[t_i, \bar{t}_i]$  associati agli eventi (nodi) del progetto; gli archi in grassetto individuano le attività critiche e i cammini critici dal nodo origine al nodo pozzo.



La seguente tabella riporta, oltre alla durata ( $c_{ij}$ ), i tempi di inizio al più presto ( $ES_{ij} = t_i$ ), di fine al più presto ( $EF_{ij} = t_i + c_{ij}$ ), di inizio al più tardi ( $LS_{ij} = \bar{t}_j - c_{ij}$ ) e di fine al più tardi ( $LF_{ij} = \bar{t}_j$ ) per ciascuna attività  $(i, j)$ , nonché il relativo slittamento  $F_{ij} = LF_{ij} - EF_{ij} = LS_{ij} - ES_{ij} = (\bar{t}_j - c_{ij}) - t_i$ . Le attività  $(i, j)$  per le quali  $F_{ij} = 0$  sono "critiche", ossia le attività C, I, L, M, O, che costituiscono gli archi del "cammino critico" (0, 4, 5, 6, 8, 9), ovvero del cammino di lunghezza massima, la cui lunghezza, pari a 48, rappresenta la durata (minima) del progetto.



Attività (i, j)	$c_{ij}$	$ES_{ij} = t_i$	$EF_{ij} = t_i + c_{ij}$	$LS_{ij} = \bar{t}_j - c_{ij}$	$LF_{ij} = \bar{t}_j$	$F_{ij} = (\bar{t}_j - c_{ij}) - t_i$
A(0, 1)	4	0	4	22	26	22
B(0, 2)	1	0	1	27	28	27
C(0, 4)	40	0	40	0	40	0
D(0, 5)	10	0	10	33	43	33
E(0, 6)	30	0	30	16	46	16
F(1, 2)	2	4	6	26	28	22
G(2, 3)	8	6	14	28	36	22
H(3, 8)	2	14	16	45	47	31
I(4, 5)	3	40	43	40	43	0
L(5, 6)	3	43	46	43	46	0
M(6, 8)	1	46	47	46	47	0
N(3, 7)	11	14	25	36	47	22
O(8, 9)	1	47	48	47	48	0
X(7, 8)	0	25	25	47	47	22
Y(3, 4)	0	14	14	40	40	26

Mentre la durata minima del progetto relativo alla sostituzione della caldaia è pari a  $t_9 = 48$  giorni, il periodo di inattività dell'impianto può essere inferiore. La durata minima di inattività dell'impianto si ottiene in particolare ritardando al massimo l'inizio delle attività A = (0, 1) e B = (0, 2). Noto che  $F_{01} = 22$  e  $F_{02} = 27$  giorni, la minima durata di inattività dell'impianto è pari a  $t_9 - \min(F_{01}, F_{02}) = 48 - 22 = 26$  giorni.