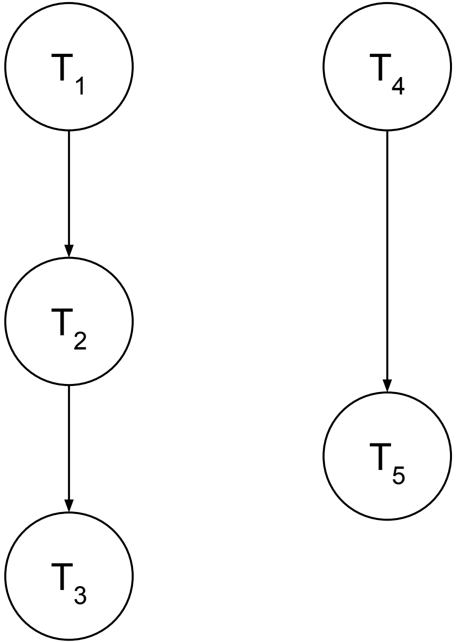
Graphe de précédence



Expression

$$(T_1\ T_2\ T_3)\ //\ (T_4\ T_5)$$

Pseudo-code

```
parbegin
  début
    T1;
    T2;
    T3
  fin
  début
    T4;
    T5
  fin
parend
```

Programme

```
T1 : X=read(5)
T2 : X=X+Z
T3 : Print Y
T4 : Z=read(7)
T5 : Y=X+Z
```

Comportements (décrits en « mots »)

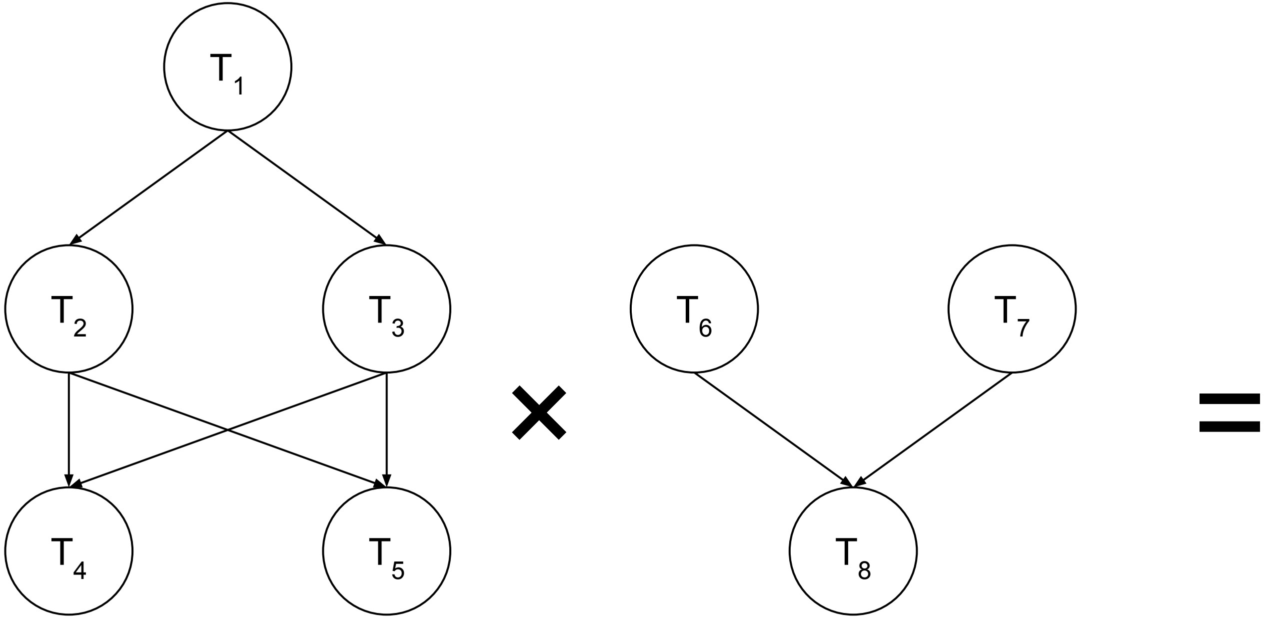
d1 f1 d4 f4 d2 f2 d5 f5 d3 f3

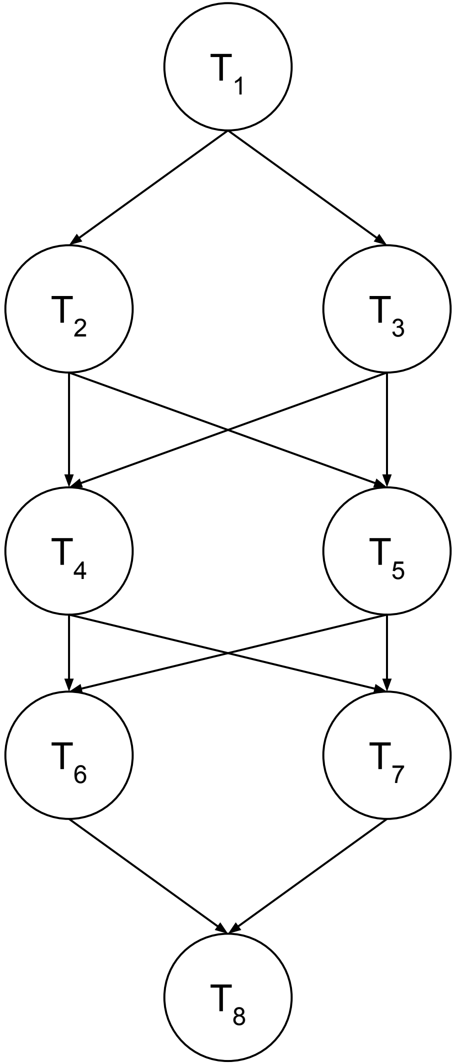
Y = ?

d1 d4 f1 f4 d2 d5 f2 f5 d3 f3

Y = ?

Système déterminé ?





Produit : combinaison séquentielle de deux systèmes de tâches
↳ Relier toutes les tâches terminales du premier, à toutes les tâches initiales du second

Exercice 2.1

1. $T_1\,((T_2\,T_3) \parallel T_4)\,T_5\,T_6$

```
début
   T1 ;
   parbegin
      début T2; T3 fin;
      T4
   parend;
   T5; T6
fin;
```

Exercice 2.1

2. $T_1 \parallel (T_2\,(T_3 \parallel T_4)\,T_5) \parallel T_6$

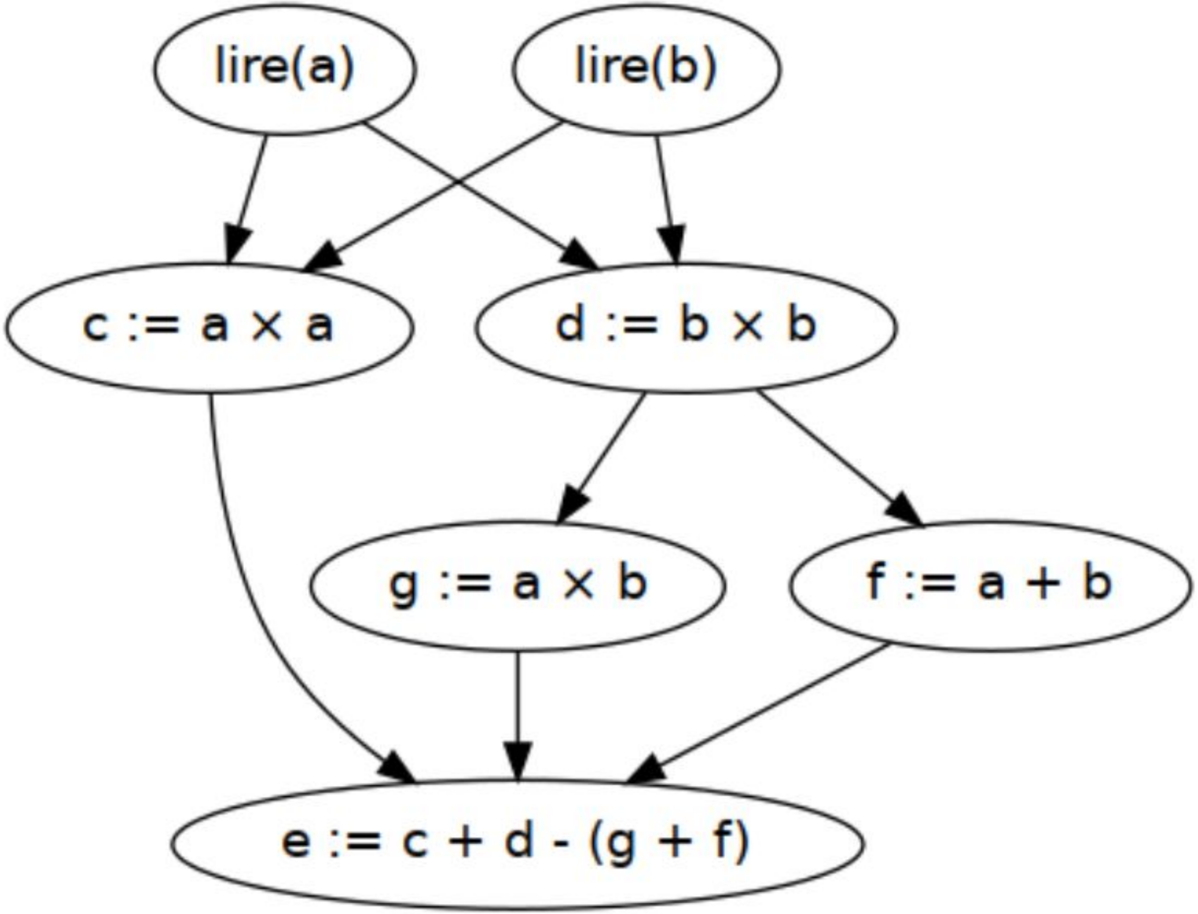
```
parbegin
   T1 ;
   début
      T2 ;
      parbegin T3 ; T4 parend;
      T5
   fin;
   T6
parend
```

Exercice 2.1

3. $(T_1\, T_2) \parallel (T_3\, T_4\, ((T_5 \parallel T_6)\, T_7))$

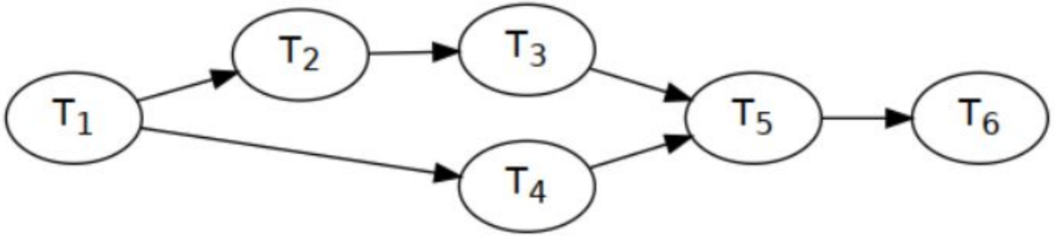
```
parbegin
  début T1 ; T2 fin
  début
    T3 ; T4 ;
    début
      parbegin T5 ; T6 parend;
      T7
    fin
  fin
parend
```

Exercice 2.2



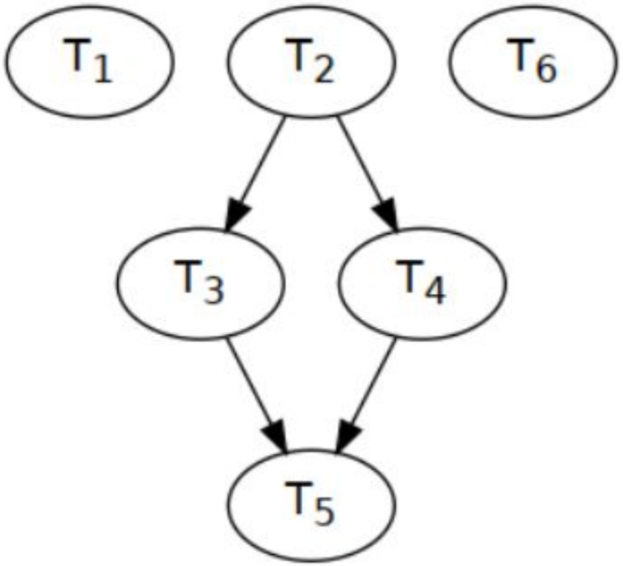
Exercice 2.2

1. $(T_1\,((T_2\,T_3) \parallel T_4)\,T_5\,T_6)$



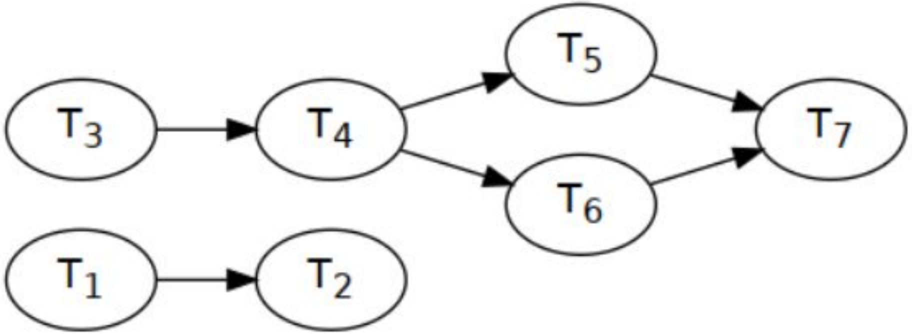
Exercice 2.2

2. $T_1 \parallel (T_2\,(T_3 \parallel T_4)\,T_5) \parallel T_6$



Exercice 2.2

3. $(T_1\,T_2) \parallel (T_3\,T_4\,((T_5 \parallel T_6)\,T_7))$



Exercice 2.3

Graphe 1 :

```
début
  T1;
  parbegin
    début
      T2; T4;
      parbegin T5; T6 parend
    fin;
    T3
  parend;
  T7
fin
```

$$T_1\,((T_2\,T_4\,(T_5 \parallel T_6)) \parallel T_3)\,T_7$$

Exercice 2.3

Graphe 2 :

```
début
   T1 ;
   parbegin
      début T2; T4 fin;
      T3
   parend;
   début
      parbegin T5; T6 parend;
      T7
   fin
fin
```

$$T_1\,((T_2\,T_4) \parallel T_3)\,((T_5 \parallel T_6)\,T_7)$$

Exercice 2.4

Graphe 1

début
  parbegin
    début
      T_1 ;
      parbegin
        début T_3 ; T_5 **fin** ;
        début T_4 ; T_6 **fin** ;
        T_2
      parend ;
    fin ;
    T_7
  parend ;
  T_8
fin

$$((T_1\,((T_3\,T_5) \parallel (T_4\,T_6) \parallel T_2)) \parallel T_7)\,T_8$$

Graphe 3

début
  T_1 ;
  parbegin T_2 ; T_3 ; T_4 **parend** ;
  parbegin T_5 ; T_6 ; T_7 **parend** ;
  T_8
fin

$$T_1\,(T_2 \parallel T_3 \parallel T_4)\,(T_5 \parallel T_6 \parallel T_7)\,T_8$$

Exercice 3.1

1. $w_1 = d_1 f_1 d_2 f_2$:

w_1	d_1	f_1	d_2	f_2		
M_1	α	α	$\alpha+\beta$	$\alpha+\beta$	$\alpha+\beta$	$V(M_1, w_1) = (\alpha, \alpha+\beta)$
M_2	β	β	β	β	$\alpha+2\beta$	$V(M_2, w_1) = (\beta, \alpha+2\beta)$

2. $w_2 = d_1 d_2 f_1 f_2$:

w_2	d_1	d_2	f_1	f_2		
M_1	α	α	α	$\alpha+\beta$	α	$V(M_1, w_2) = (\alpha, \alpha+\beta, \alpha)$
M_2	β	β	β	β	$\alpha+\beta$	$V(M_2, w_2) = (\beta, \alpha+\beta)$

3. $w_3 = d_1 d_2 f_2 f_1$:

w_3	d_1	d_2	f_2	f_1		
M_1	α	α	α	α	$\alpha+\beta$	$V(M_1, w_3) = (\alpha, \alpha+\beta)$
M_2	β	β	β	$\alpha+\beta$	β	$V(M_2, w_3) = (\beta, \alpha+\beta, \beta)$

4. $w_4 = d_2 d_1 f_1 f_2$:

w_4	d_2	d_1	f_1	f_2		
M_1	α	α	α	$\alpha+\beta$	α	$V(M_1, w_4) = (\alpha, \alpha+\beta, \alpha)$
M_2	β	β	β	β	$\alpha+\beta$	$V(M_2, w_4) = (\beta, \alpha+\beta)$

5. $w_5 = d_2 d_1 f_2 f_1$:

w_5	d_2	d_1	f_2	f_1		
M_1	α	α	α	α	$\alpha+\beta$	$V(M_1, w_5) = (\alpha, \alpha+\beta)$
M_2	β	β	β	$\alpha+\beta$	β	$V(M_2, w_5) = (\beta, \alpha+\beta, \beta)$

6. $w_6 = d_2 f_2 d_1 f_1$:

w_6	d_2	f_2	d_1	f_1		
M_1	α	α	α	α	$2\alpha+\beta$	$V(M_1, w_6) = (\alpha, 2\alpha+\beta)$
M_2	β	β	$\alpha+\beta$	$\alpha+\beta$	$\alpha+\beta$	$V(M_2, w_6) = (\beta, \alpha+\beta)$

Exercice 3.2

1. $w_1 = d_1 f_1 d_2 f_2$:

w_1	d_1	f_1	d_2	f_2		
M_1	α	α	$\alpha+\beta$	$\alpha+\beta$	$\alpha+\beta$	$V(M_1, w_1) = (\alpha, \alpha+\beta)$
M_2	β	β	β	β	$\alpha+2\beta$	$V(M_2, w_1) = (\beta, \alpha+2\beta)$

2. $w_2 = d_1 d_2 f_1 f_2$:

w_2	d_1	d_2	f_1	f_2		
M_1	α	α	α	$\alpha+\beta$	$\alpha+\beta$	$V(M_1, w_2) = (\alpha, \alpha+\beta)$
M_2	β	β	β	β	$\alpha+\beta$	$V(M_2, w_2) = (\beta, \alpha+\beta)$

3. $w_3 = d_1 d_2 f_2 f_1$:

w_3	d_1	d_2	f_2	f_1		
M_1	α	α	α	α	$\alpha+\beta$	$V(M_1, w_3) = (\alpha, \alpha+\beta)$
M_2	β	β	β	$\alpha+\beta$	$\alpha+\beta$	$V(M_2, w_3) = (\beta, \alpha+\beta)$

4. $w_4 = d_2 d_1 f_1 f_2$:

w_4	d_2	d_1	f_1	f_2		
M_1	α	α	α	$\alpha+\beta$	$\alpha+\beta$	$V(M_1, w_4) = (\alpha, \alpha+\beta)$
M_2	β	β	β	β	$\alpha+\beta$	$V(M_2, w_4) = (\beta, \alpha+\beta)$

5. $w_5 = d_2 d_1 f_2 f_1$:

w_5	d_2	d_1	f_2	f_1		
M_1	α	α	α	α	$\alpha+\beta$	$V(M_1, w_5) = (\alpha, \alpha+\beta)$
M_2	β	β	β	$\alpha+\beta$	$\alpha+\beta$	$V(M_2, w_5) = (\beta, \alpha+\beta)$

6. $w_6 = d_2 f_2 d_1 f_1$:

w_6	d_2	f_2	d_1	f_1		
M_1	α	α	α	α	$2\alpha+\beta$	$V(M_1, w_6) = (\alpha, 2\alpha+\beta)$
M_2	β	β	$\alpha+\beta$	$\alpha+\beta$	$\alpha+\beta$	$V(M_2, w_6) = (\beta, \alpha+\beta)$

Exercice 3.2

$E = \{T_1, \ldots, T_{100}\}$

Processus : exécution d'un programme

$P_1 : S = (\{T_1, \ldots, T_{100}\}, <)$
$P_2 : S = (\{T_1, \ldots, T_{100}\}, <)$
$P_3 : S = (\{T_1, \ldots, T_{100}\}, <)$

Pour atteindre −149, valeur minimale de `compte` :

$d_1\ d_1\ f_1 \ldots d_{50}\ f_{50}\ d_1\ f_1 \ldots d_{50}\ f_{50}\ f_1\ d_{51}\ d_2\ f_2 \ldots d_{50}\ f_{50}\ f_{51}\ d_{52}\ f_{52} \ldots d_{100}\ f_{100}\ d_{51}\ f_{51} \ldots d_{100}\ f_{100}\ d_{51}\ f_{51} \ldots d_{100}\ f_{100}$

Pour atteindre 149, valeur maximale de `compte` :

$d_1\ f_1 \ldots d_{50}\ f_{50}\ d_1\ f_1 \ldots d_{50}\ f_{50}\ d_1\ f_1 \ldots d_{50}\ d_{51}\ f_{51} \ldots d_{100}\ f_{100}\ d_{51}\ f_{51} \ldots d_{99}\ f_{99}\ f_{50}\ d_{100}\ d_{51}\ f_{51} \ldots d_{100}\ f_{100}\ f_{100}$

Conditions de Bernstein

T1 et T2 sont non-interférentes entre elles si :

- $T1 < T2$ ou $T2 < T1$
- ou : $L1 \cap E2 = L2 \cap E1 = E1 \cap E2 = \varnothing$

Un système est déterminé si toute paire de tâches est non-interférente.

Exercice 4.1

Une condition suffisante pour que le système soit déterminé est que toute paire de tâches soit non interférente. Comme les tâches connectées par une flèche sont explicitement ordonnées et donc par définition non intérférentes, on vérifiera l'absence d'interférence pour les paires de tâches n'ayant pas de relation de précédence entre elles :
T_3 et T_4,
T_3 et T_6,
T_5 et T_6.
Plus concrètement, on vérifie les conditions de Bernstein pour les domaines de lecture et d'écriture de ces tâches :

$$L_3 \cap E_4 = E_3 \cap L_4 = E_3 \cap E_4 = \emptyset,$$

$$L_3 \cap E_6 = E_3 \cap L_6 = E_3 \cap E_6 = \emptyset,$$

$$L_5 \cap E_6 = E_5 \cap L_6 = E_5 \cap E_6 = \emptyset.$$

On constate l'absence d'interférences entre ces trois paires de tâches ; le système S est donc déterminé.

Parallélisme maximal : réduire un système de tâches pour avoir le meilleur taux d'utilisation du processeur.

Le système réduit et le système initial doivent donner les mêmes résultats ; c.a.d qu'ils doivent être équivalents :
$S(E, <)$ et $S'(E, <')$ sont équivalents si

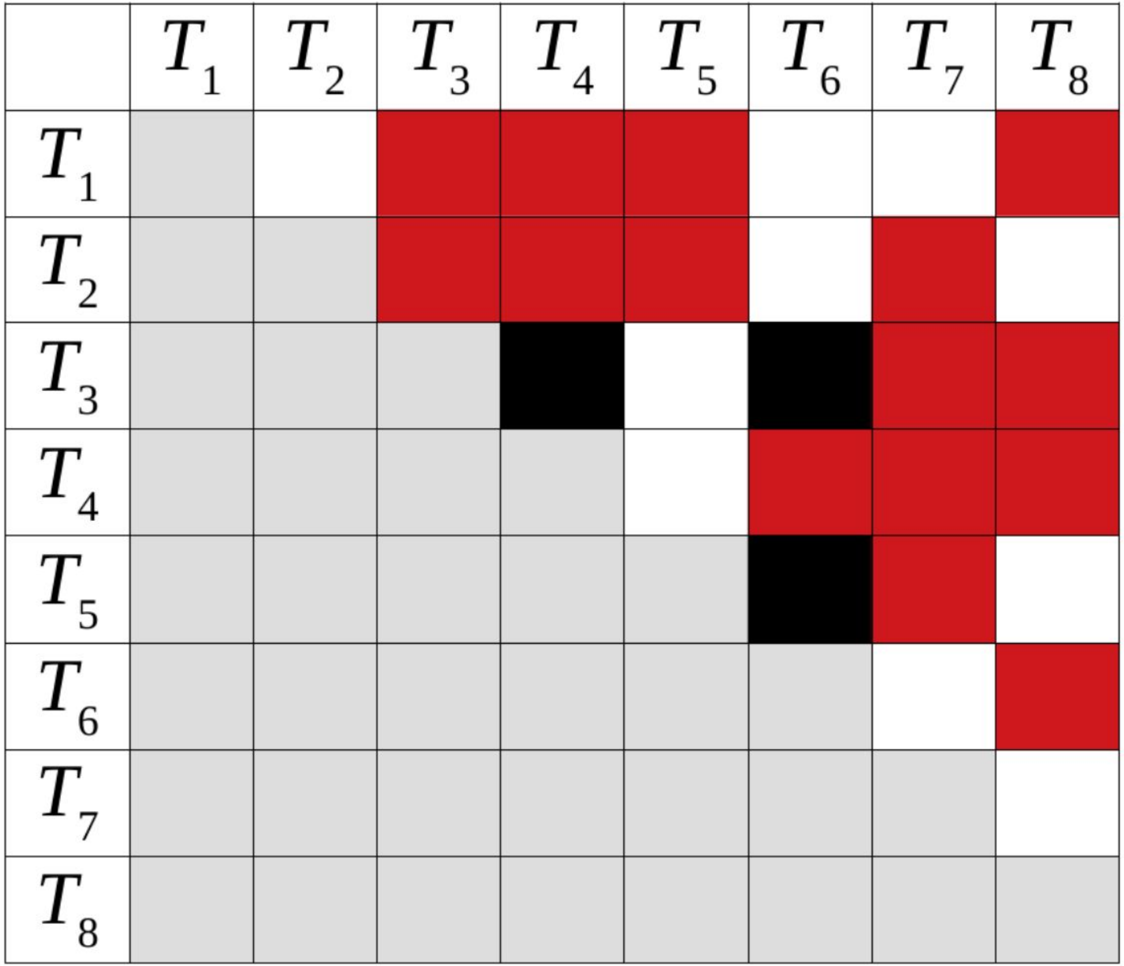
- ils sont tous les deux déterminés
- et si l'état final M_f de la mémoire est le même pour un même état M_0

Un système de tâche sera de parallélisme maximal si la suppression de tout arc (T, T') du graphe (c.a.d la suppression d'une relation de précédence T < T' ou T' < T), entraîne l'interférence des tâches (T, T'), donc le non-déterminisme du système.

La technique de construction de ce système minimal consiste à éliminer tous les arcs redondants.

Exercice 4.1

	M_1	M_2	M_3	M_4	M_5
T_1	L	L	E		
T_2	L			E	
T_3	E		L	L	
T_4			L	L	E
T_5		E		L	
T_6					L/E
T_7	L	L		L/E	
T_8	L		L		E



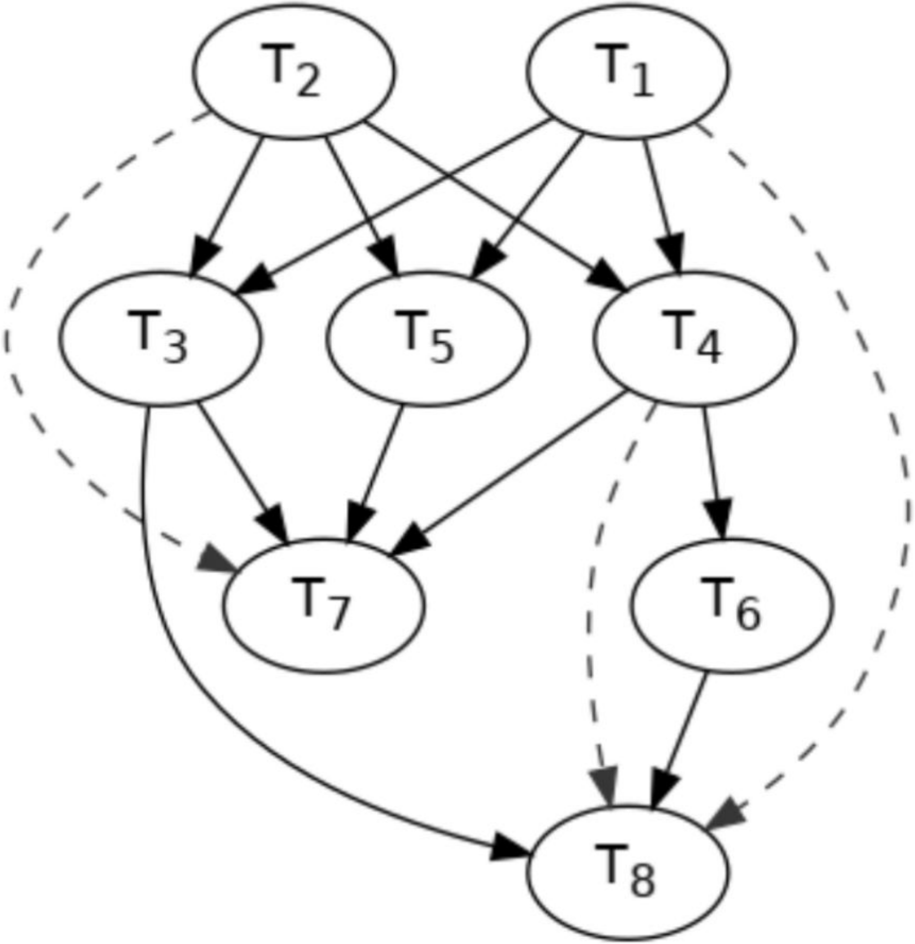
Paires de tâches non-considérées dans le calcul, car non connectées dans le graphe initial.

Paires de tâches en interférence sur les accès à la mémoire.

Exercice 4.1

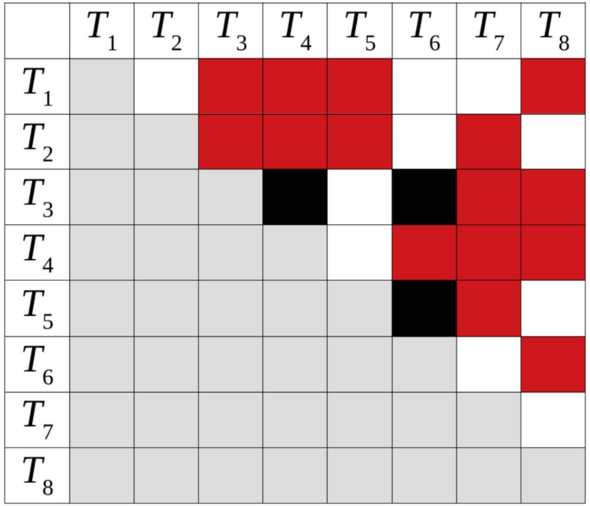
Ajout d'arcs pour chaque 🟥

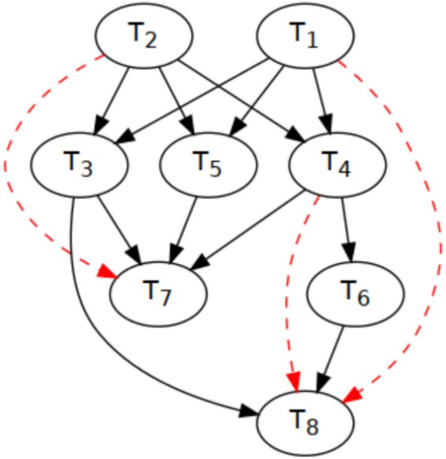
Puis, éliminations des arcs redondants.



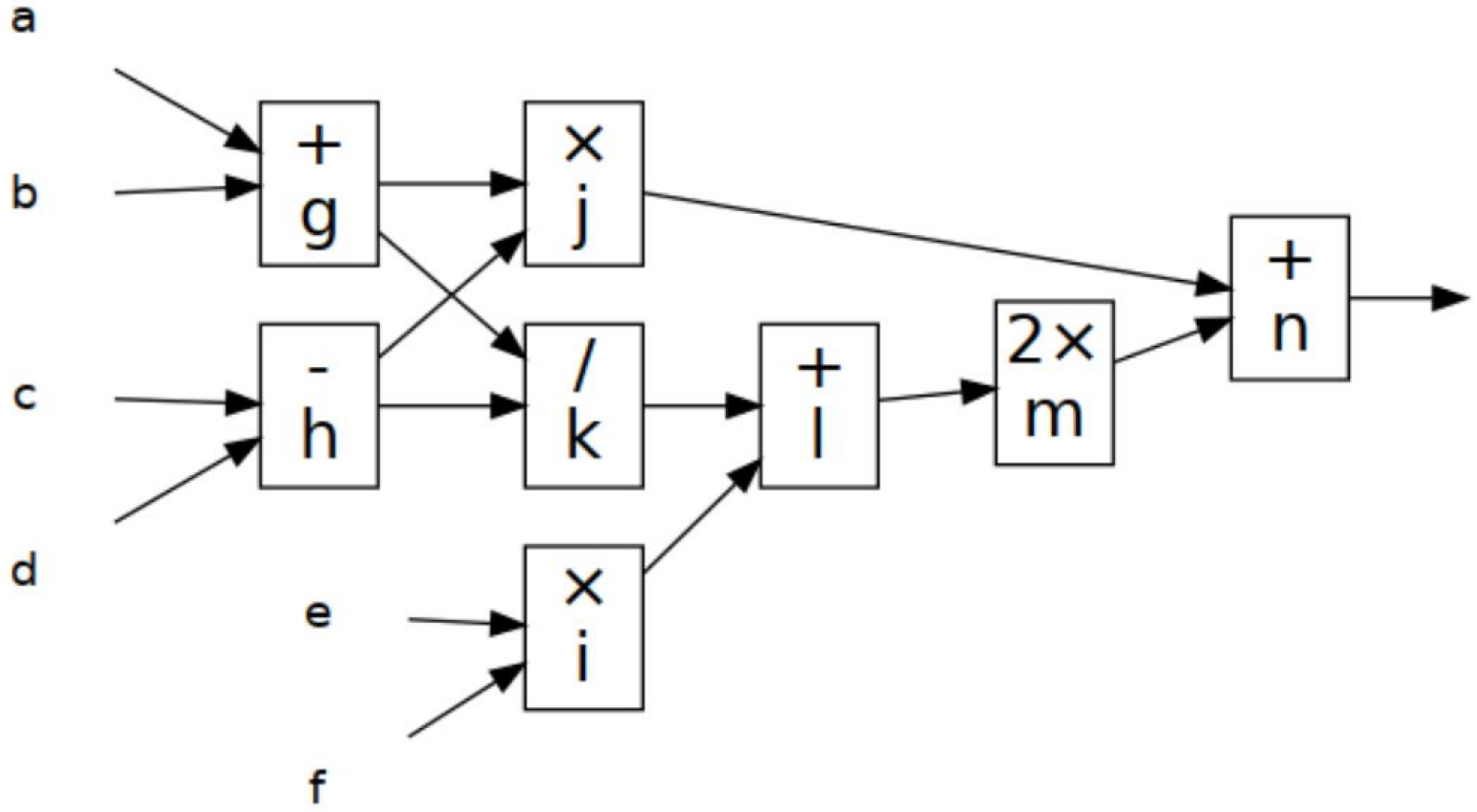
Exercice 4.1

	M_1	M_2	M_3	M_4	M_5
T_1	L	L	E		
T_2	L			E	
T_3	E		L	L	
T_4			L	L	E
T_5		E		L	
T_6					L/E
T_7	L	L		L/E	
T_8	L		L		E





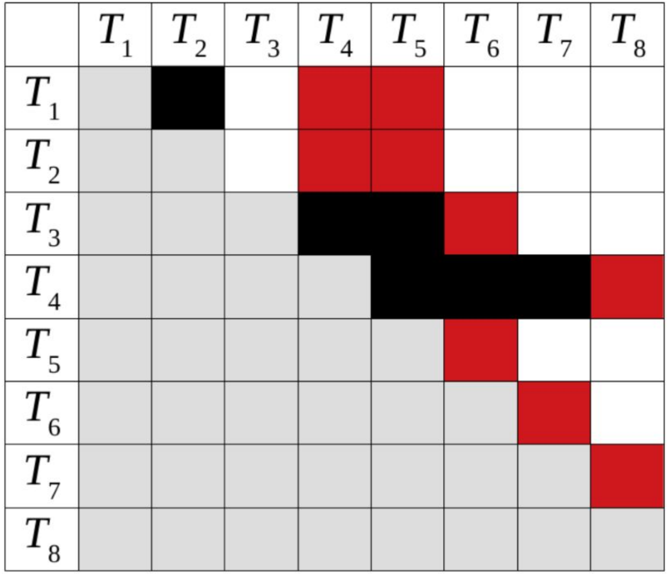
Exercice 4.2



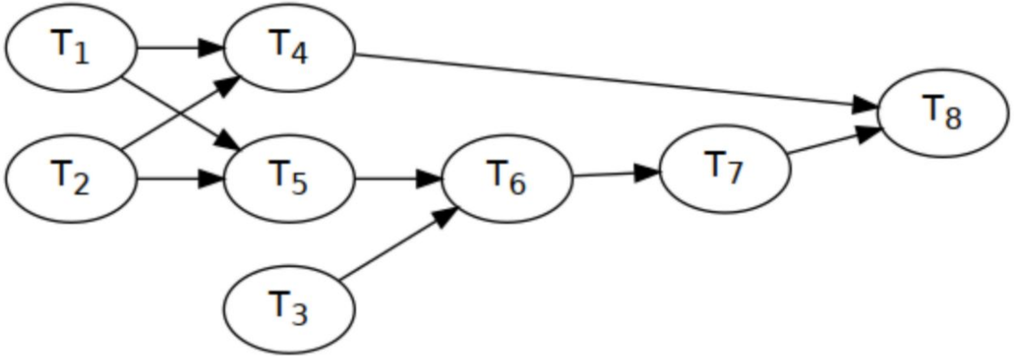
Exercice 4.2

Programme	Tâche	L_i	E_i
début			
`g := a + b;`	T_1	a, b	g
`h := c - d;`	T_2	c, d	h
`i := e × f;`	T_3	e, f	i
`j := g × h;`	T_4	g, h	j
`k := g/h;`	T_5	g, h	k
`l := k + i;`	T_6	k, i	l
`m := 2 × l;`	T_7	l	m
`n := j + m`	T_8	j, m	n
fin			

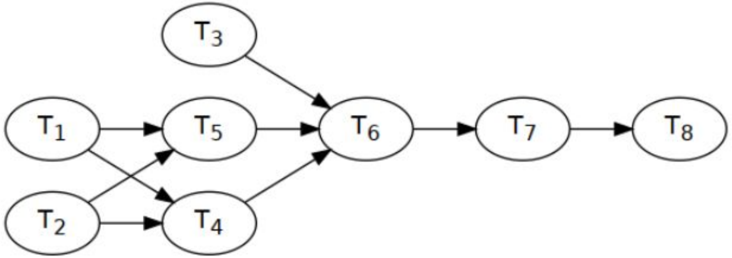
	a	b	c	d	e	f	g	h	i	j	k	l	m	n
T_1	L	L					E							
T_2			L	L				E						
T_3					L	L			E					
T_4							L	L		E				
T_5							L	L			E			
T_6									L		L	E		
T_7												L	E	
T_8										L			L	E



Exercice 4.2



Exercice 4.2



Ce graphe de précédence peut être décrit avec l'expression

$$(((T_1 \parallel T_2)\,(T_4 \parallel T_5)) \parallel T_3)\,T_6\,T_7\,T_8.$$

Le programme parallèle correspondant s'écrit de la façon suivante :

```
début
  parbegin
    début
      parbegin g := a + b; h := c - d parend;
      parbegin j := g × h; k := g/h parend
    fin;
    i := e × f;
  parend;
  l := k + i;
  m := 2 × l;
  n := j + m;
fin
```