

Ordonnancements simples.









Jérémy Rouot

e-mail: jeremy.rouot@yncrea.fr

bureau 332



Vocabulaire sur l'orientation

<i>Graphe non orienté</i>		<i>Graphe orienté</i>	
	arête	arc	
	chaîne	chemin	
	cycle	circuit	
	arbre	arborescence	



Lien entre Connexité et Arbres couvrants

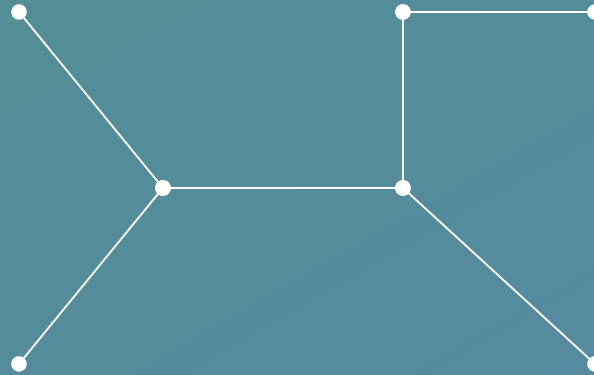




On appelle ***arbre*** un graphe *connexe* et *sans cycle*.

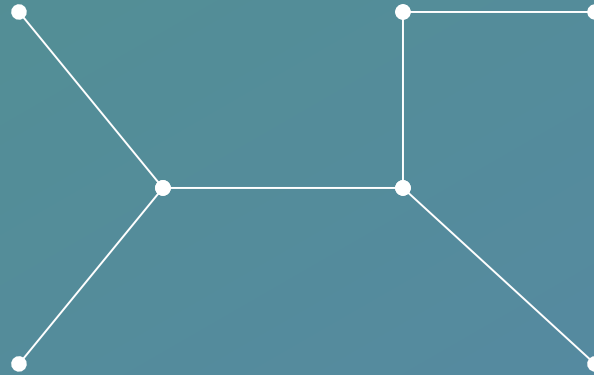


On appelle ***arbre*** un graphe *connexe* et *sans cycle*.



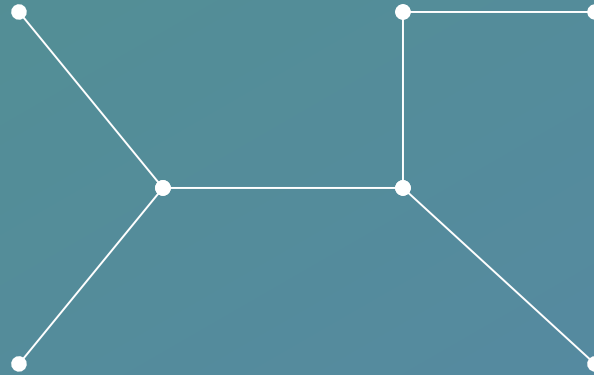
On appelle ***arbre*** un graphe *connexe* et *sans cycle*.

un arbre



On appelle ***arbre*** un graphe *connexe* et *sans cycle*.

un arbre

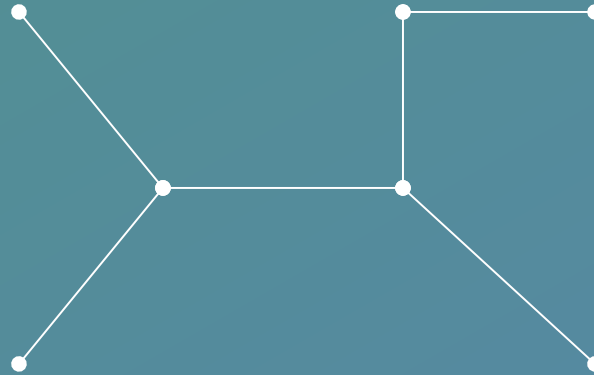


Un graphe *sans cycle* est une ***forêt***.

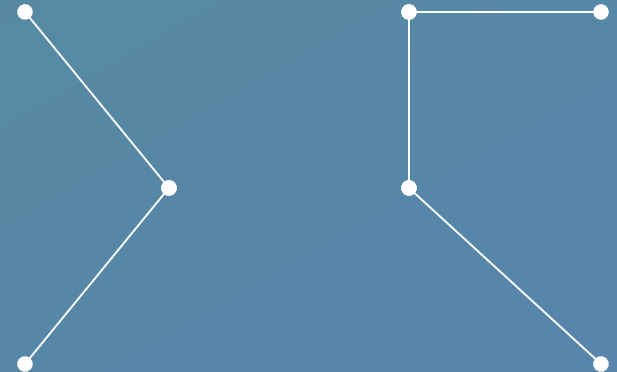


On appelle ***arbre*** un graphe *connexe* et *sans cycle*.

un arbre

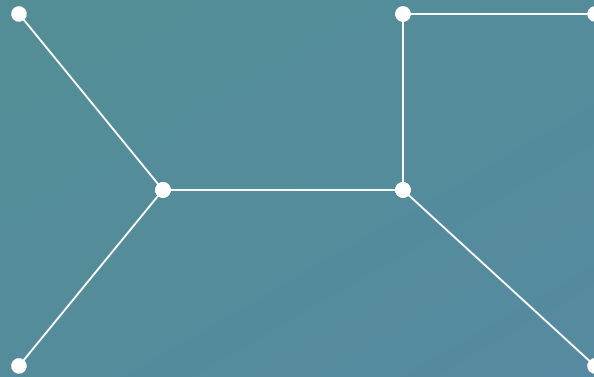


Un graphe *sans cycle* est une ***forêt***.

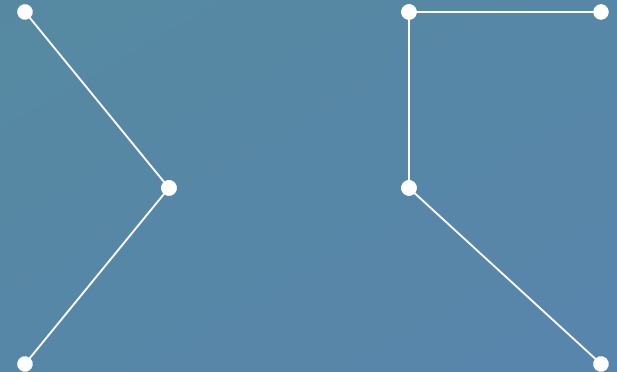


On appelle **arbre** un graphe *connexe* et *sans cycle*.

un arbre

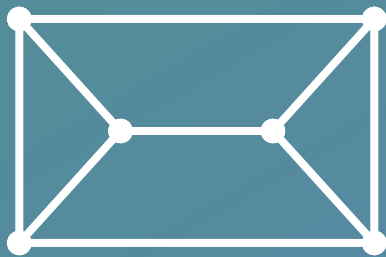


Un graphe *sans cycle* est une **forêt**.

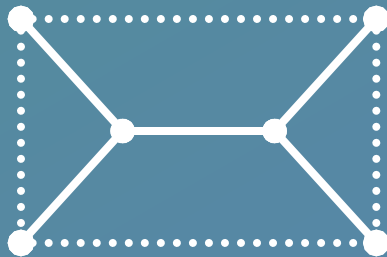


une forêt de 2 arbres

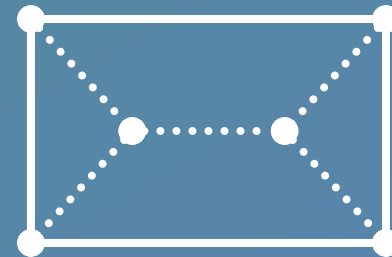




graphe G



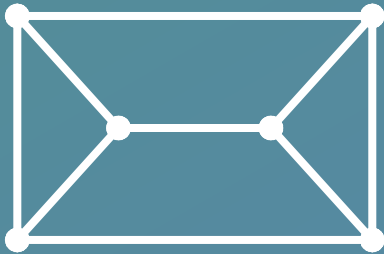
un arbre couvrant



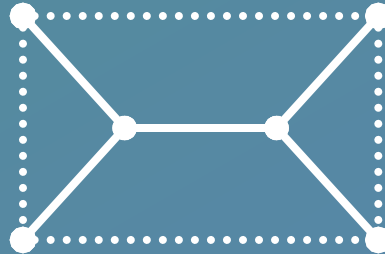
co-arbre associé



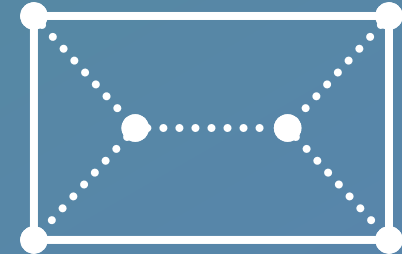
Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.



graphe G



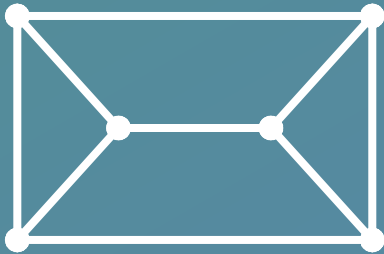
un arbre couvrant



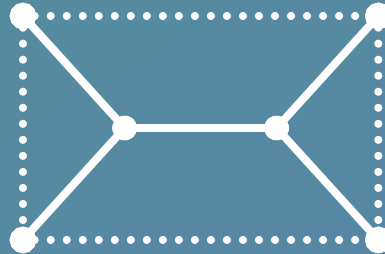
co-arbre associé



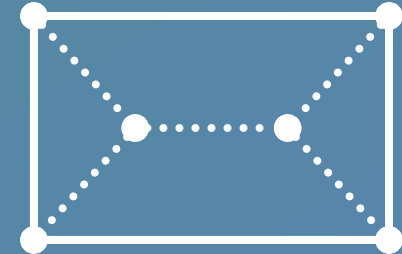
Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.



graphe G



un arbre couvrant

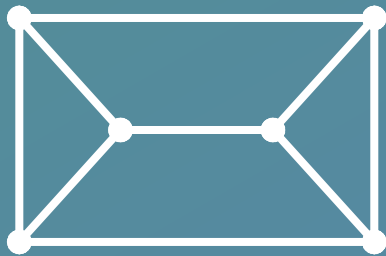


co-arbre associé

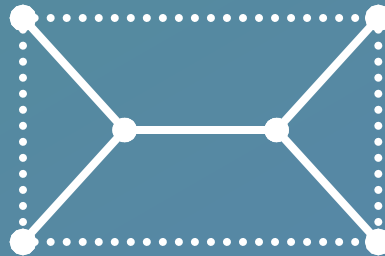


Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.

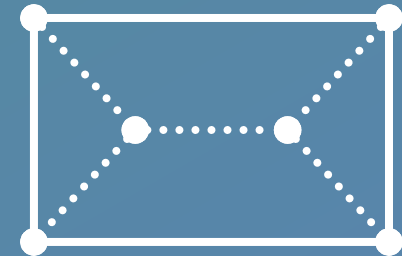
Un tel arbre est dit **couvrant** (car il couvre tous les sommets du graphe) ou **arbre du graphe** G .



graphe G



un arbre couvrant



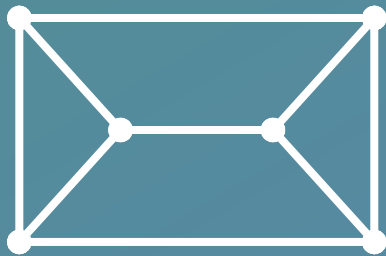
co-arbre associé



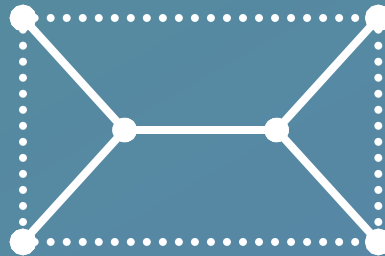
Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.

Un tel arbre est dit **couvrant** (car il couvre tous les sommets du graphe) ou **arbre du graphe** G .

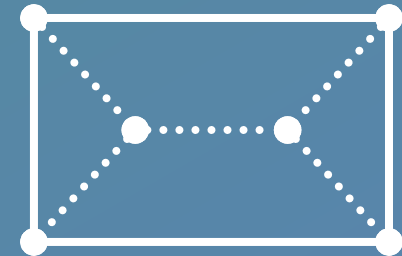
Son complémentaire par rapport à G , c'est-à-dire le graphe $K=(X,E\setminus T)$, est appelé **co-arbre associé** à A .



graphe G



un arbre couvrant



co-arbre associé

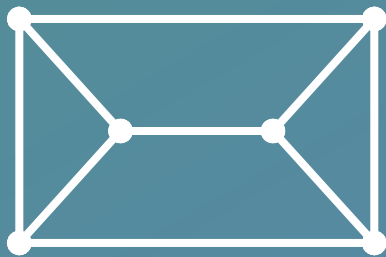


Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.

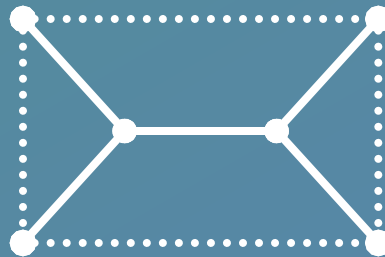
Un tel arbre est dit **couvrant** (car il couvre tous les sommets du graphe) ou **arbre du graphe** G .

Son complémentaire par rapport à G , c'est-à-dire le graphe $K=(X,E\setminus T)$, est appelé **co-arbre associé** à A .

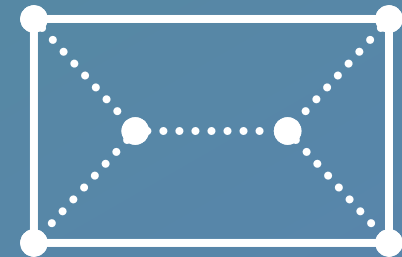
Plus généralement, on appelle co-arbre de G tout graphe partiel de G dont le complémentaire est un arbre couvrant de G .



graphe G



un arbre couvrant



co-arbre associé





L'arbre de poids minimum d'un graphe connexe.



L'arbre de poids minimum d'un graphe connexe.

Soit $G=(X;E)$ un graphe *connexe* d'ordre n et $p: E \rightarrow \mathbb{R}$ une fonction qui associe à toute arête de G un poids réel. Pour chaque arbre $A=(X;T)$ de G on définit son poids



L'arbre de poids minimum d'un graphe connexe.

Soit $G=(X;E)$ un graphe *connexe* d'ordre n et $p: E \rightarrow \mathbb{R}$ une fonction qui associe à toute arête de G un poids réel. Pour chaque arbre $A=(X;T)$ de G on définit son poids

$$p(A) = \sum p(e).$$



L'arbre de poids minimum d'un graphe connexe.

Soit $G=(X;E)$ un graphe *connexe* d'ordre n et $p: E \rightarrow \mathbb{R}$ une fonction qui associe à toute arête de G un poids réel. Pour chaque arbre $A=(X;T)$ de G on définit son poids

$$p(A) = \sum p(e).$$

On veut déterminer un arbre couvrant de G de poids minimum.



L'arbre de poids minimum d'un graphe connexe.

Soit $G=(X;E)$ un graphe *connexe* d'ordre n et $p: E \rightarrow \mathbb{R}$ une fonction qui associe à toute arête de G un poids réel. Pour chaque arbre $A=(X;T)$ de G on définit son poids

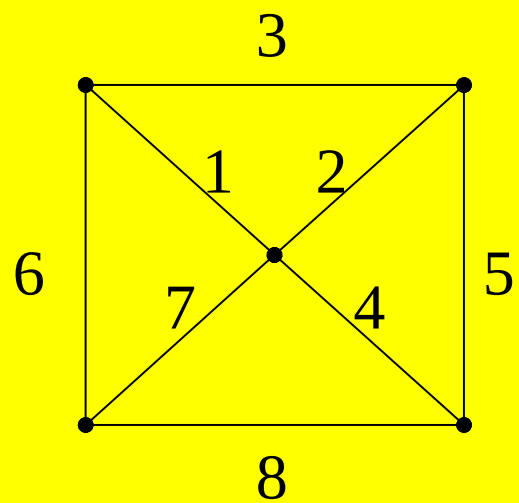
$$p(A) = \sum p(e).$$

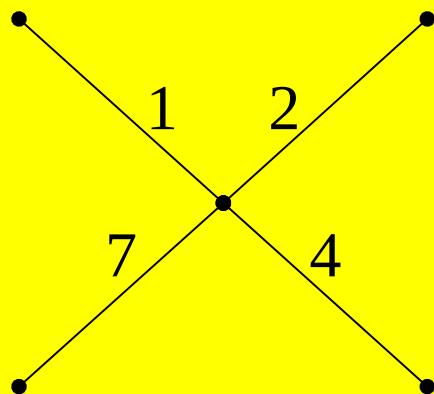
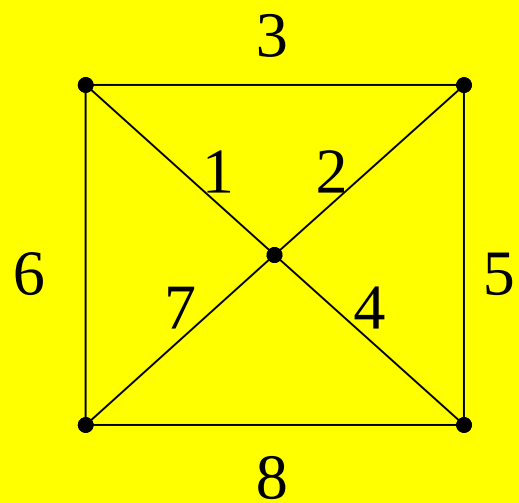
On veut déterminer un arbre couvrant de G de poids minimum.

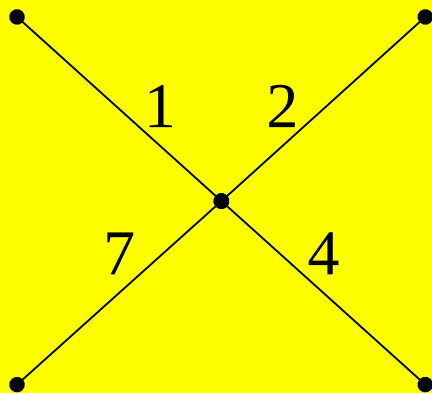
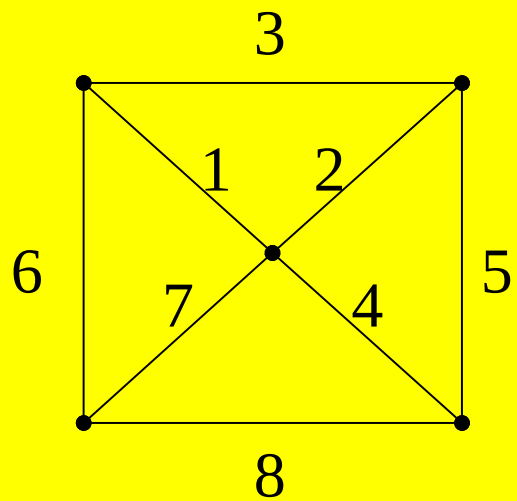
(Le problème de maximisation peut être résolu par le remplacement de la fonction p par $-p$.)





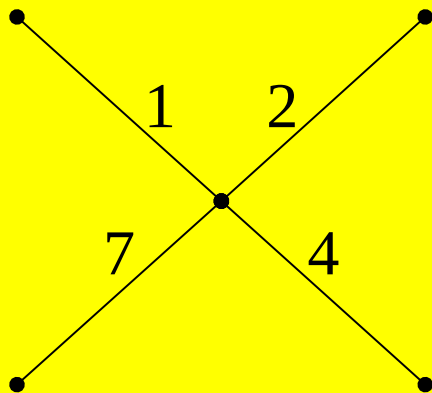
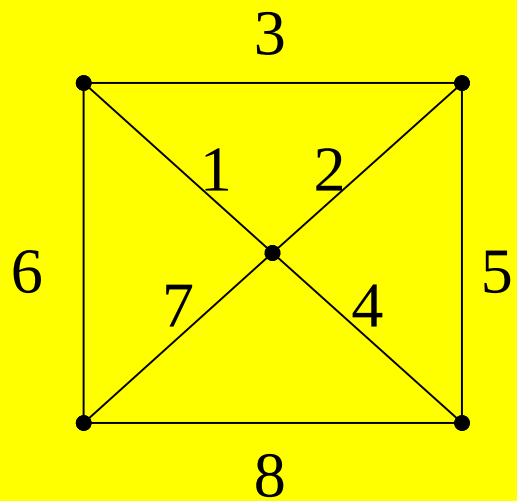




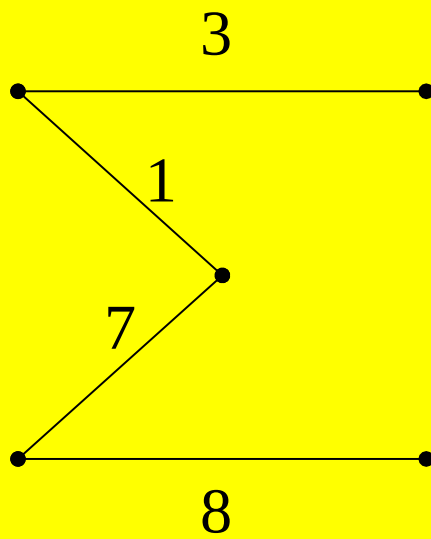


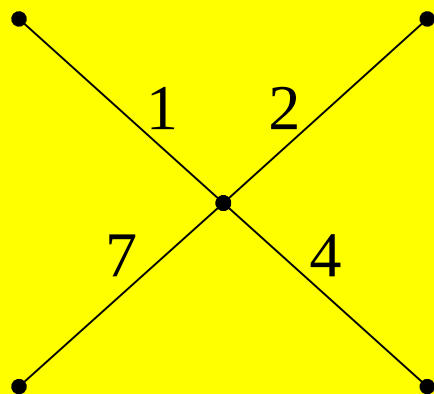
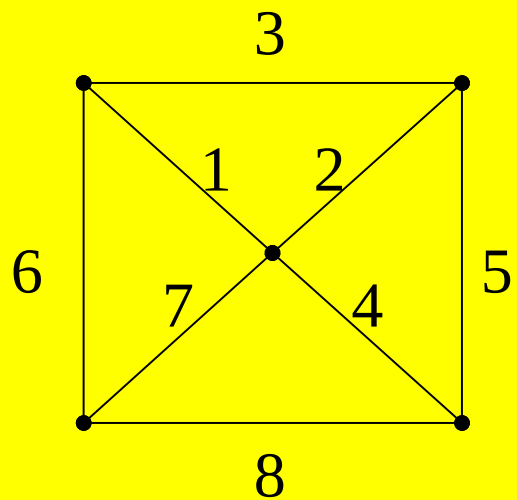
$$p(X)=14$$



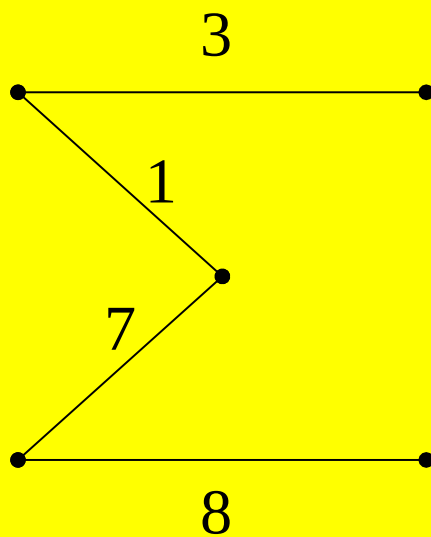


$$p(X)=14$$



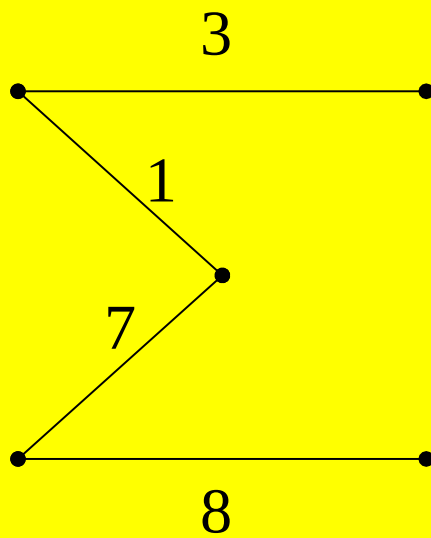
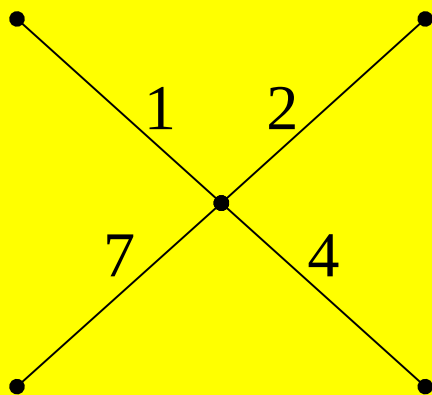
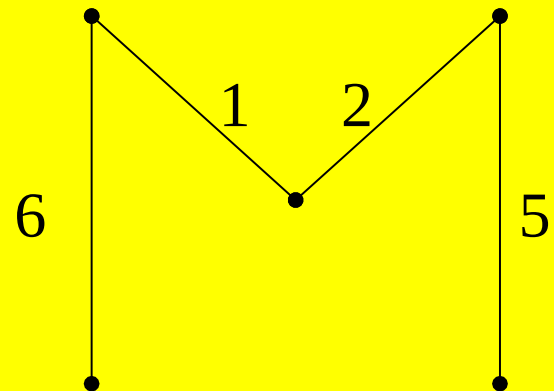
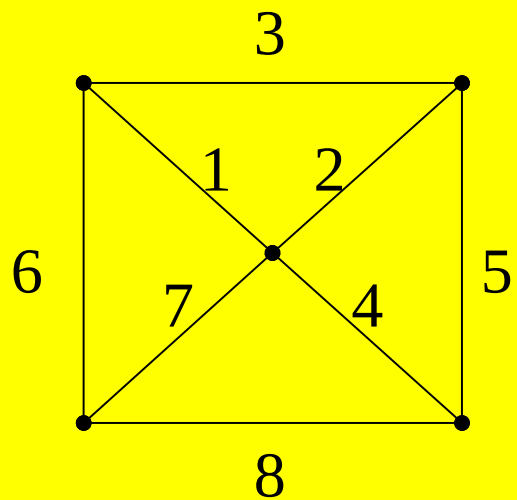


$$p(X)=14$$



$$p(\Sigma)=19$$

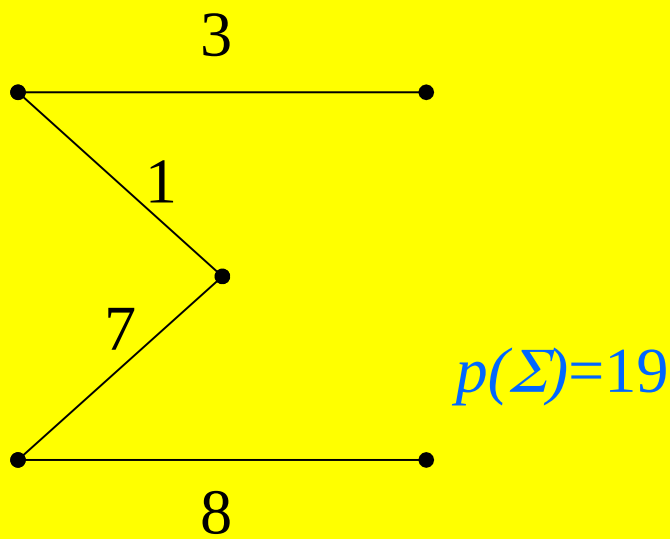
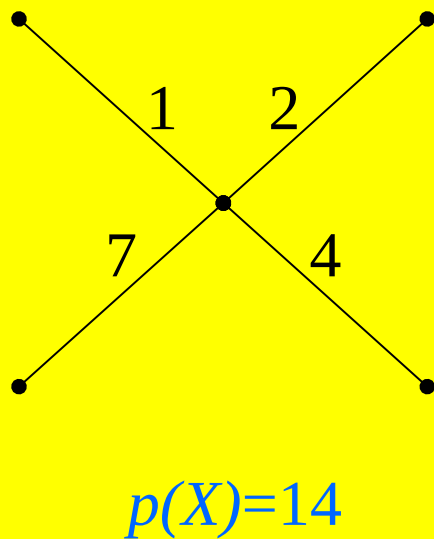
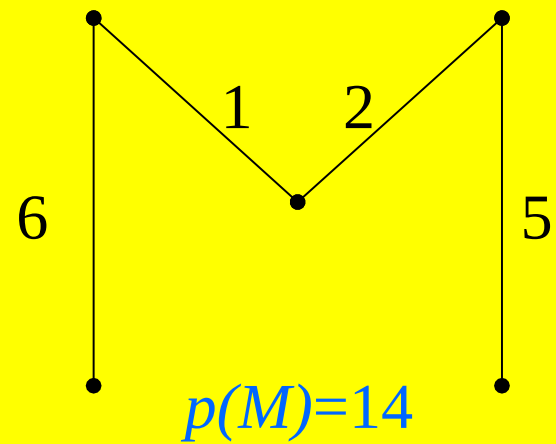
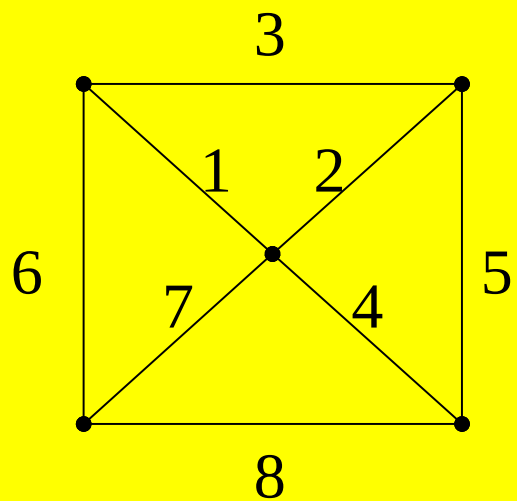




$p(\Sigma)=19$

$p(X)=14$





Algorithme de **Kruskal**:

1° **Trier** E pour obtenir la liste (e_1, e_2, \dots, e_m) telle que :

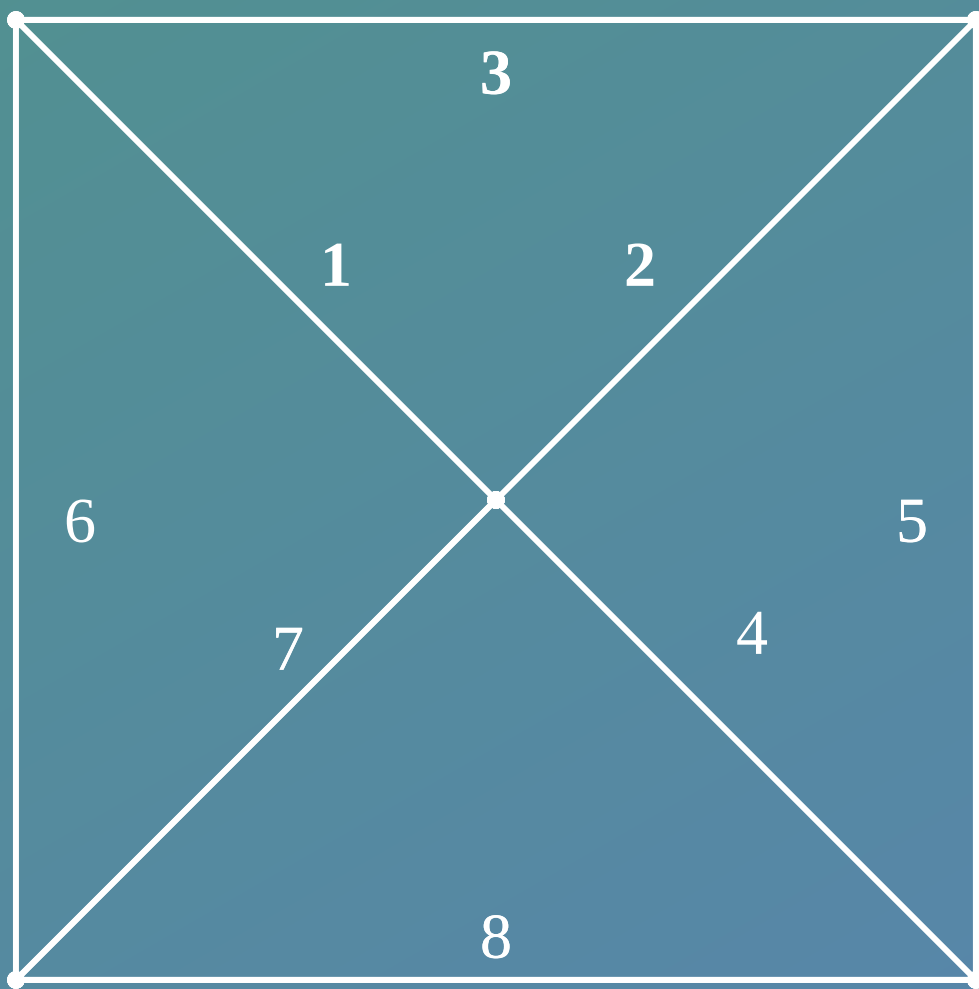
$$p(e_i) \leq p(e_{i+1}) \quad \text{pour } i=1, 2, \dots, m-1.$$

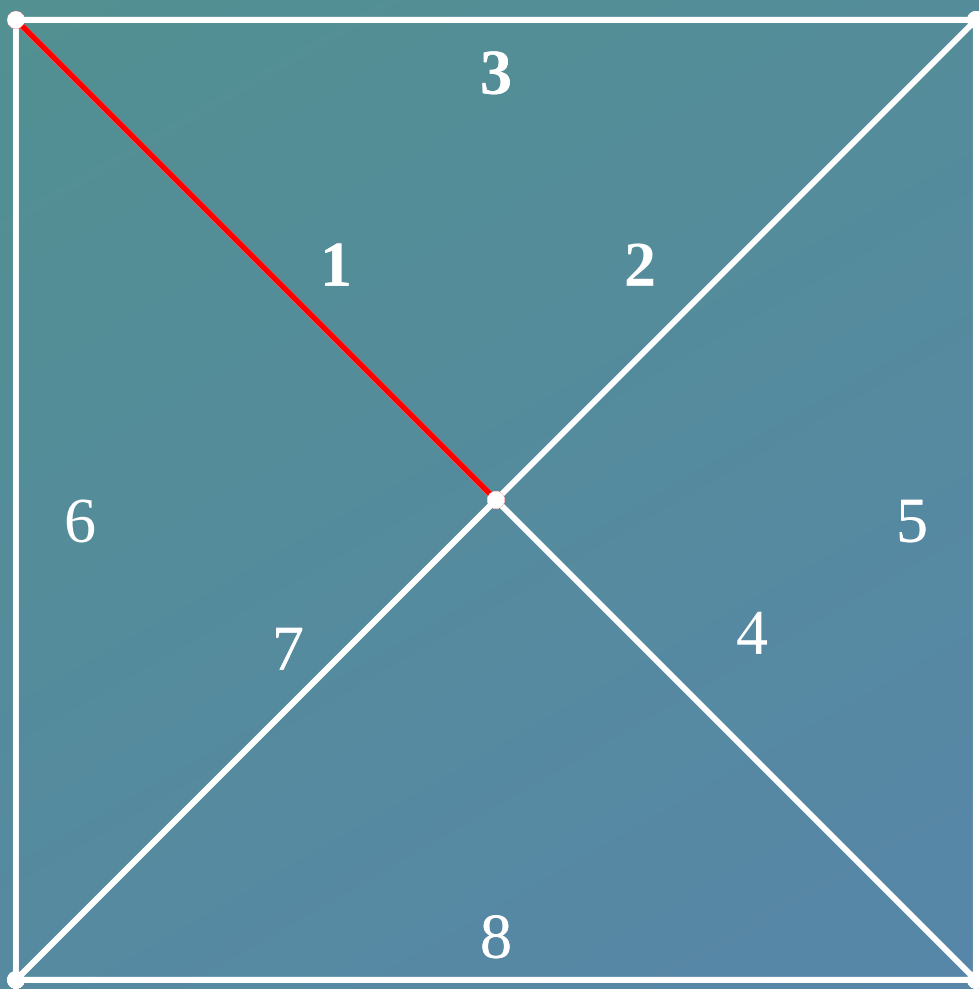
2° **Construire** la séquence: $T_1 = \emptyset$; $T_{i+1} = T_i \cup \{e_k\}$

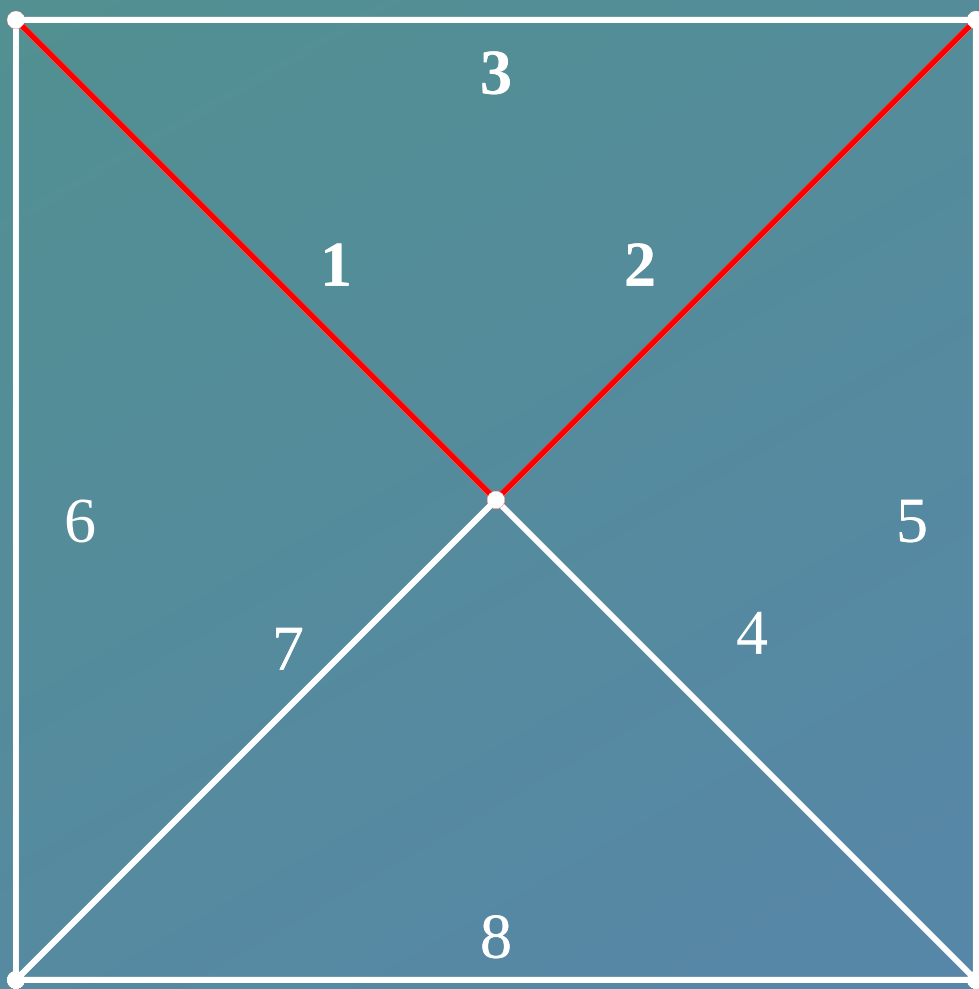
où $k = \min\{j; T_i + e_j \text{ est sans cycle}\}$.

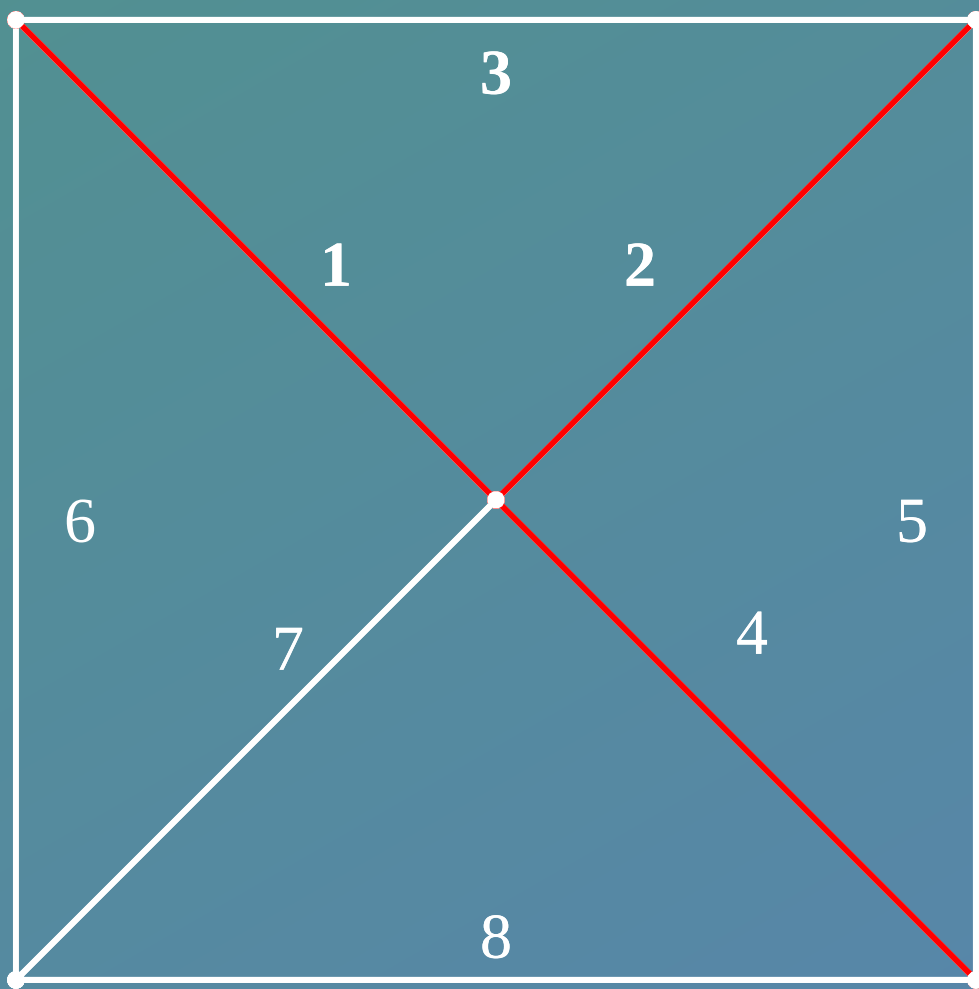
$A_n = (X; T_n)$ est un arbre de poids minimum.

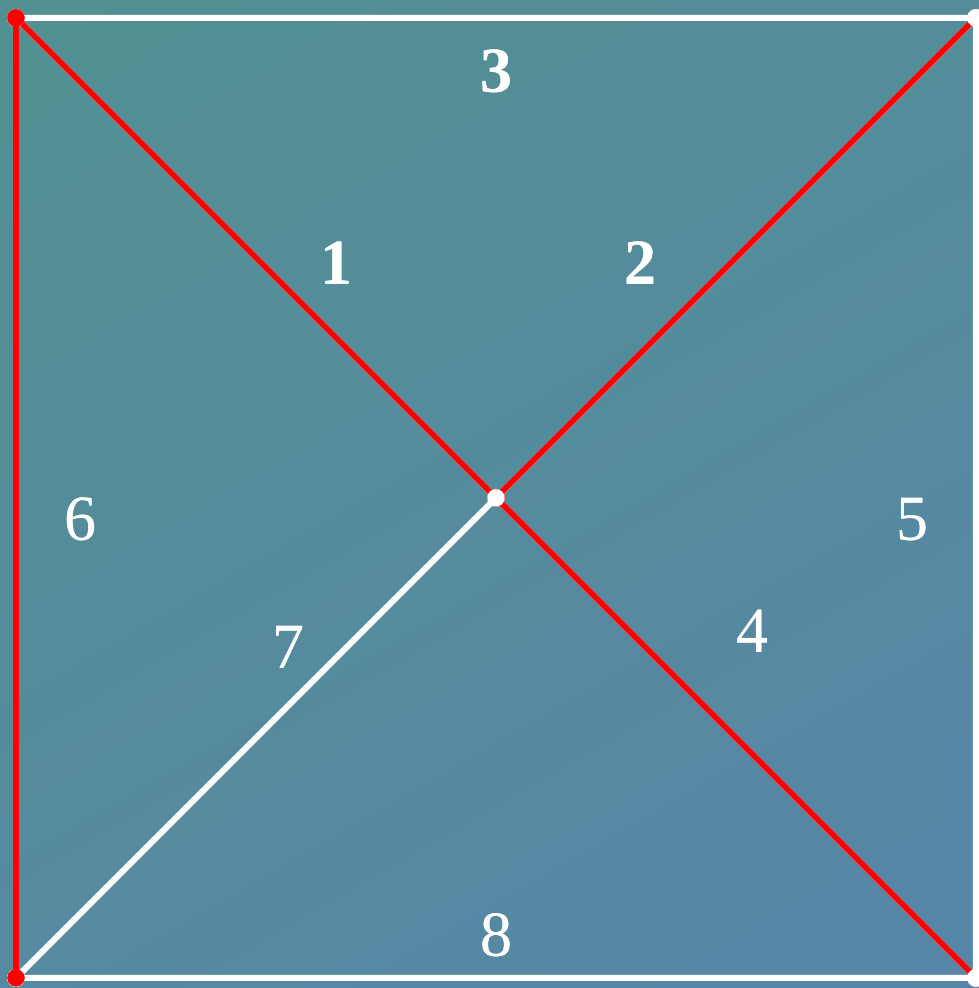




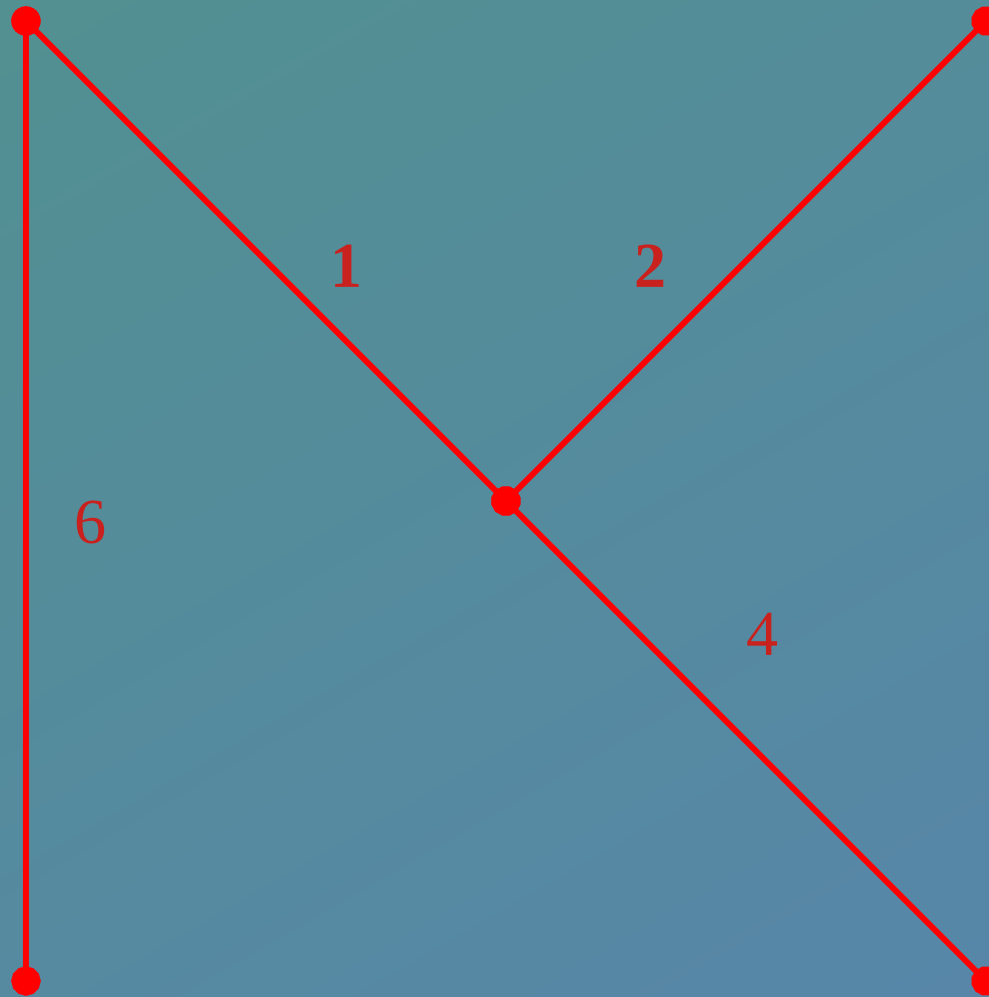








Arborescence couvrante de poids minimum



Ordonnancements.



Applications

on doit fabriquer une pièce sur une machine ;

une opération doit être réalisée par un chirurgien ;

les sous-programmes d'un programme informatique doivent être effectués par des microprocesseurs parallèles ;

la tonte de l'herbe sur l'accotement d'une route doit être réalisée par des tondeuses ou des épareuses.

Questions

Quelle tâche sur quel processeur (affectation, allocation, etc.)? Dans quel ordre réaliser les tâches sur les processeurs ?

Il faut préciser l'objectif, comme par exemple, minimiser une date de fin d'une tâche, minimiser la somme des retards (c'est-à-dire des dépassements des dates limites) etc.

La théorie de l'ordonnancement propose des modèles, principes et techniques de solution des nombreux problèmes de ce genre.



Pour un projet, composé de plusieurs tâches élémentaires, on cherche un planning (ou calendrier) d'exécution des tâches (c'est-à-dire un ensemble de dates affectées à chaque tâche, qui respecte les durées des tâches et les contraintes de précédence), qui minimise la durée totale du projet (c'est-à-dire le temps entre le début et la fin du projet).



Plus précisément on va donc s'intéresser à la **durée totale** minimum et, pour chaque tâche x :

- la **date au plus tôt**, notée $\pi(x)$, qui est la date minimum à laquelle on peut démarrer l'exécution de la tâche x compte tenu de toutes les données et contraintes du problème (par convention, le projet démarre toujours au moment 0);
- la **date au plus tard**, notée $\eta(x)$, qui est la date limite à laquelle on doit commencer l'exécution de la tâche x sous peine de retarder l'exécution totale du projet ;
- la **marge (libre)** représente le délai dont on dispose pour démarrer la tâche en respectant la durée totale minimum; on a évidemment $m(x) = \eta(x) - \pi(x)$.

Les tâches qui ont une marge nulle seront appelées **tâches critiques**. Tous retard pris dans l'exécution d'une tâche critique entraîne un retard dans la réalisation du projet. Il est donc intéressant d'établir une liste des tâches critiques.



Exemple : La construction traditionnelle d'une maison individuelle commence par l'établissement des fondations. Supposons que ce projet se décompose en cinq tâches élémentaires:

T - terrassement - durée 3 jours;

G - mise en place de la grue - durée 1 jour;

R - branchements aux réseaux d'eau et EDF - durée 2 jours;

B – coulage d'une dalle de béton - durée 3 jours;

S - installation de la fosse septique - durée 4 jours.

Les tâches **T**, **G** et **R** peuvent démarrer tout de suite, par contre on ne peut installer la fosse septique ni couler la dalle de béton qu'après avoir effectué les travaux de terrassement. Il est aussi évident que la pose d'une dalle de béton nécessite de l'eau et aussi la présence de la grue qui, pour fonctionner, a besoin de l'électricité.

Le problème est de proposer un calendrier d'exécution des tâches respectant les contraintes d'antériorité (contraintes de précédence) et permettant de réaliser l'ensemble des tâches en un temps minimum.



tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T

diagrammes de GANTT





tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T

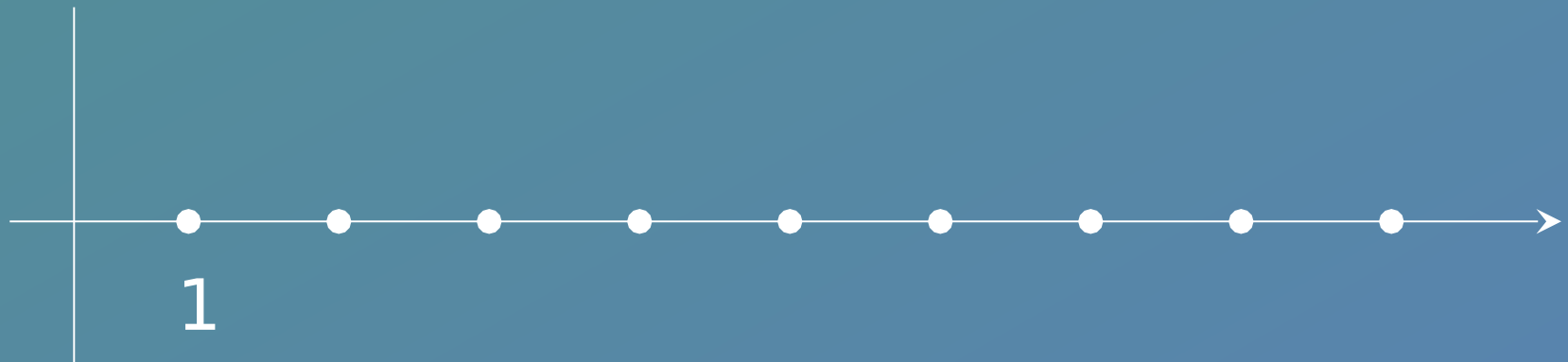


tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



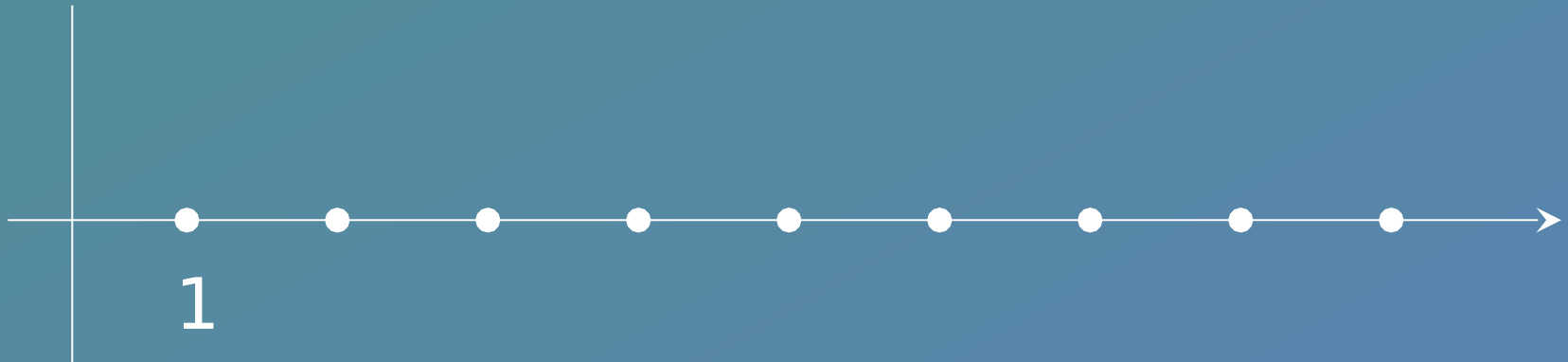


tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



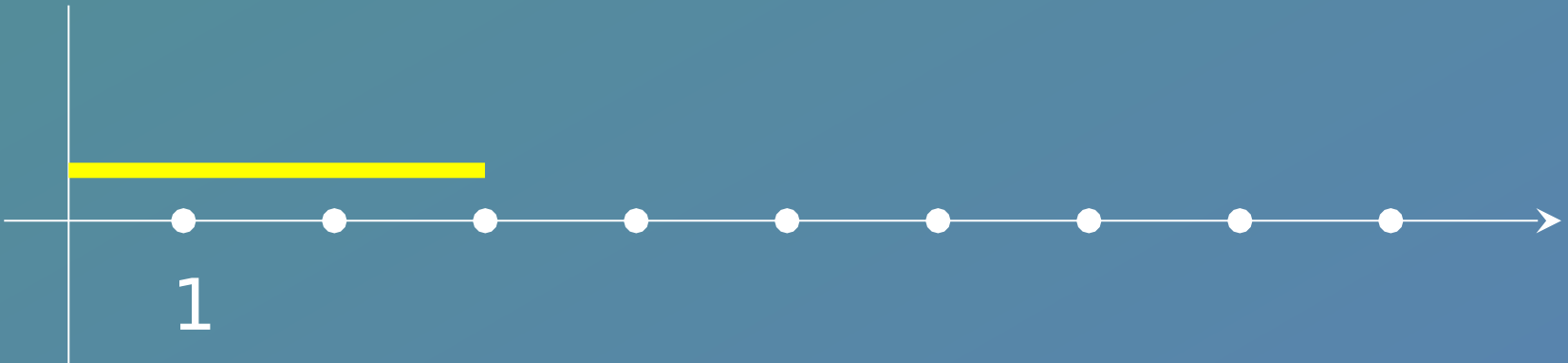


tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



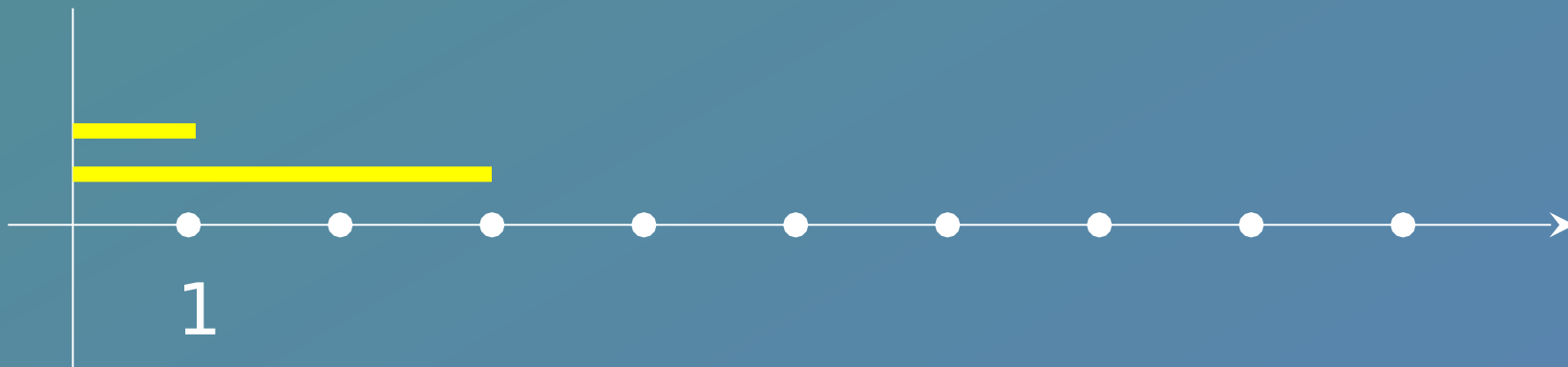


tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T

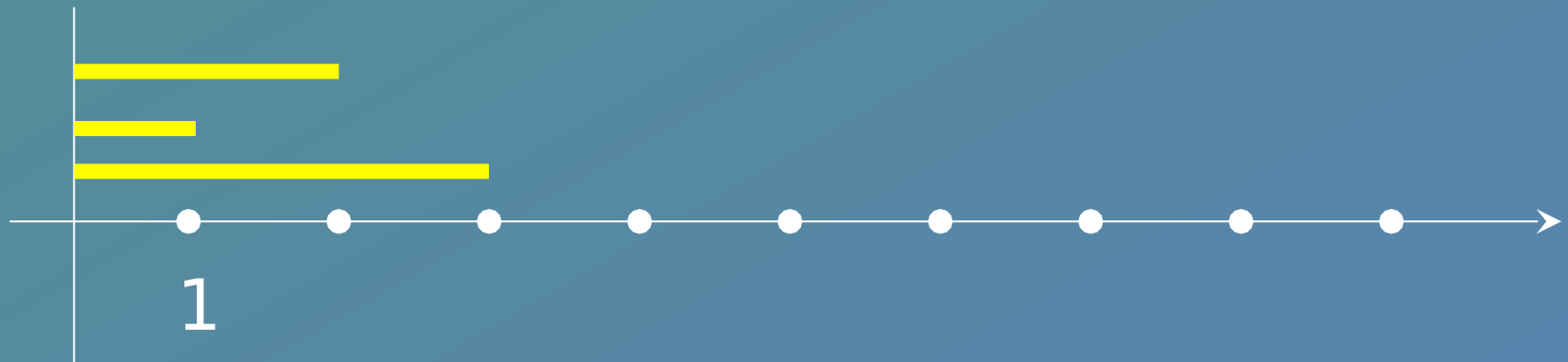




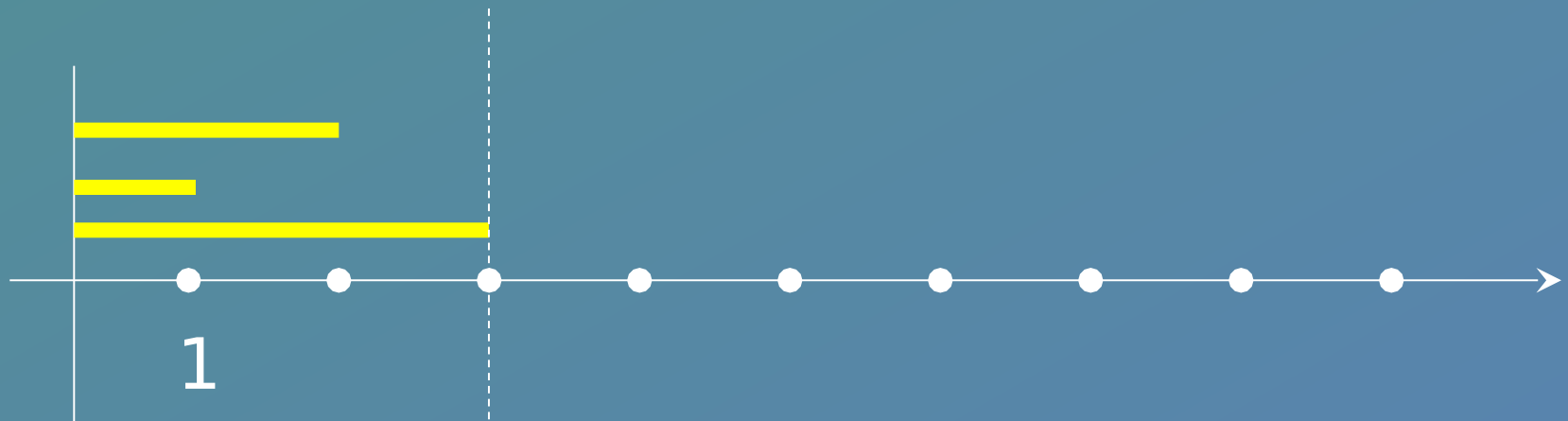
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



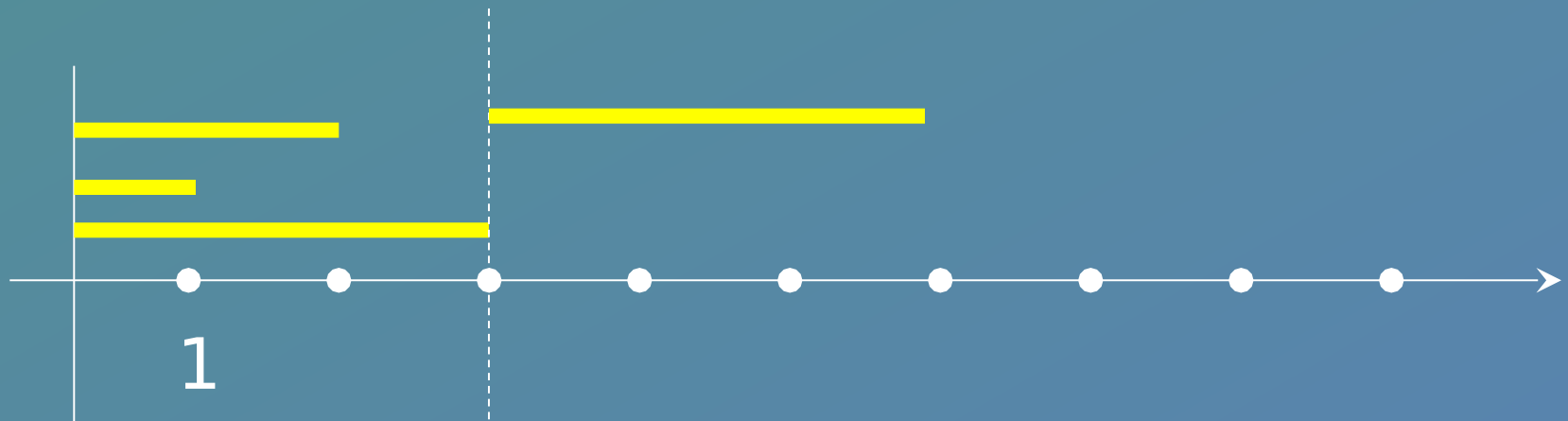
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



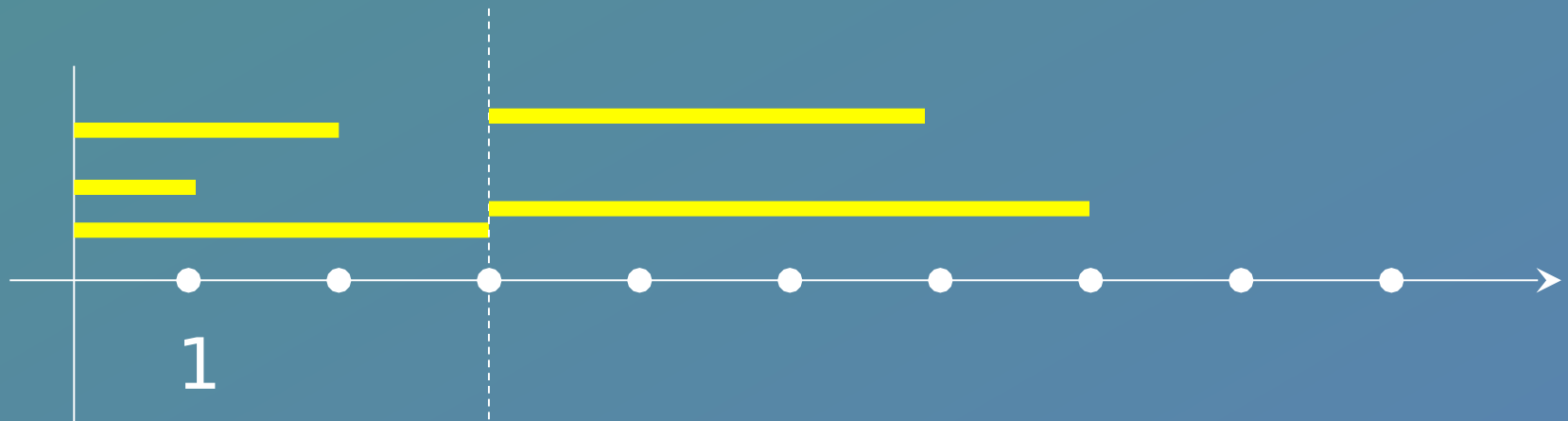
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



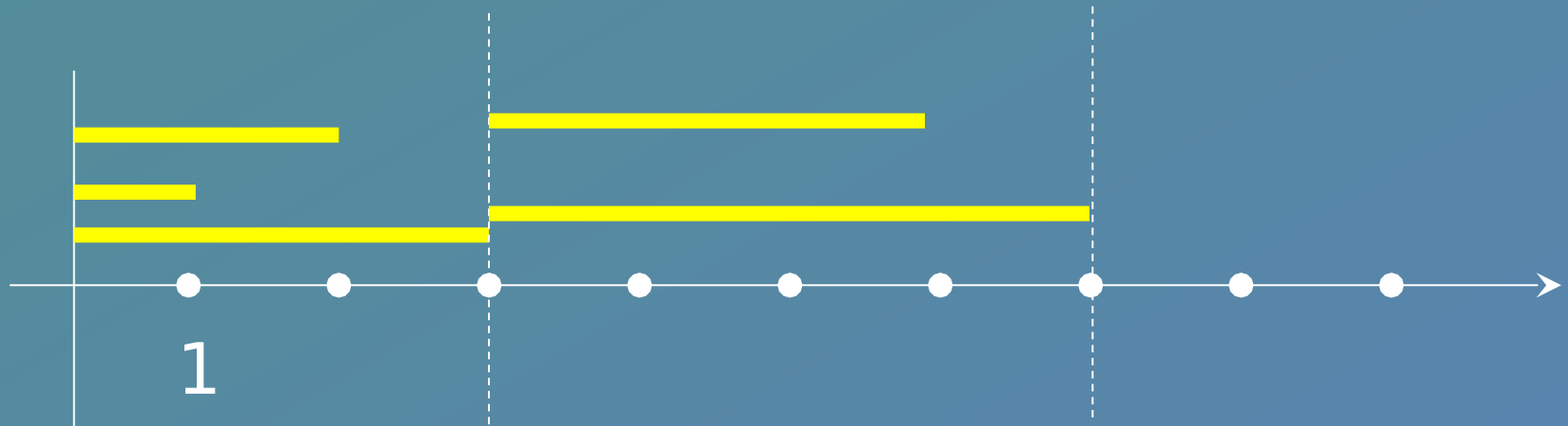
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



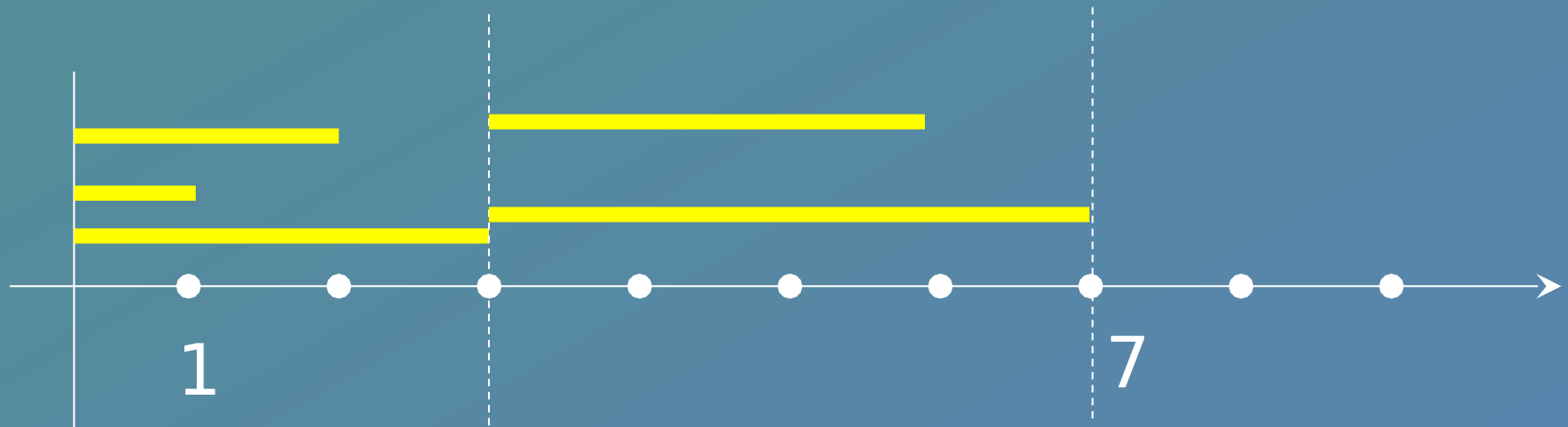
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



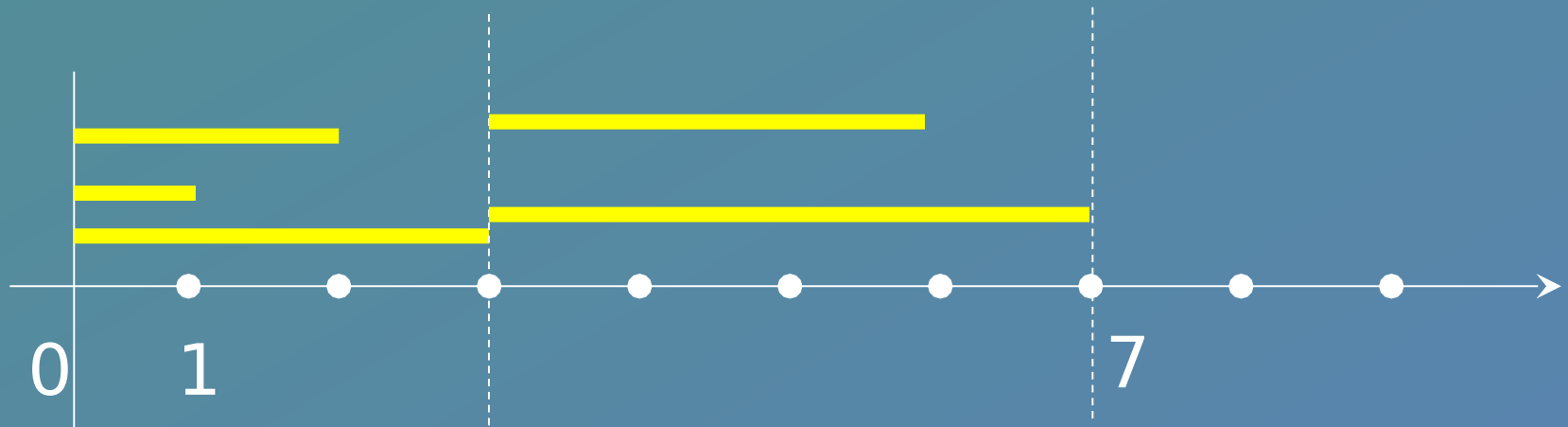
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



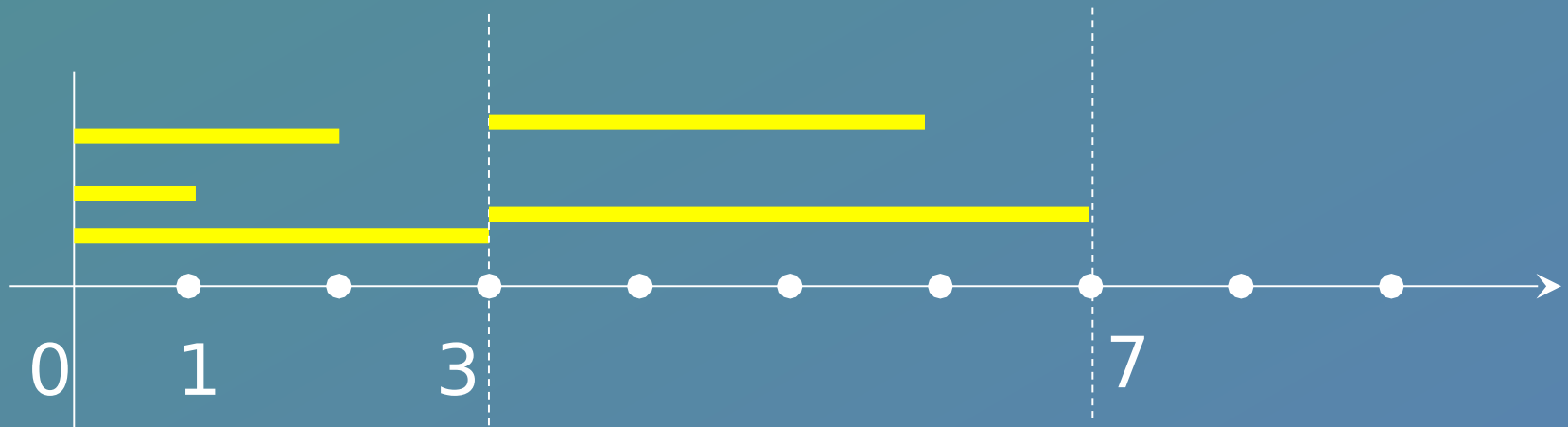
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



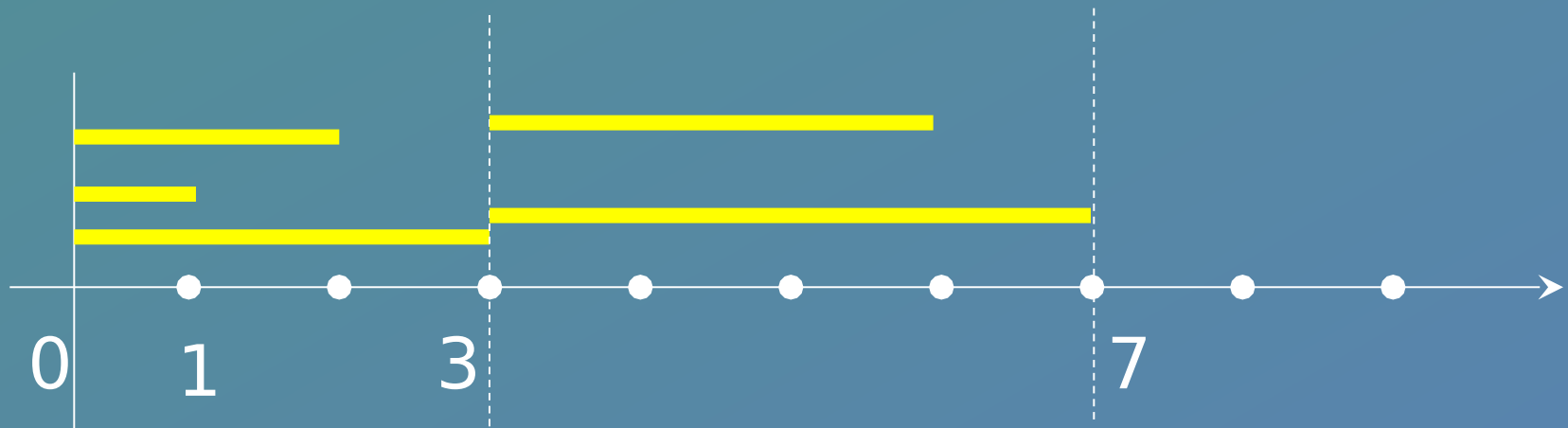
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



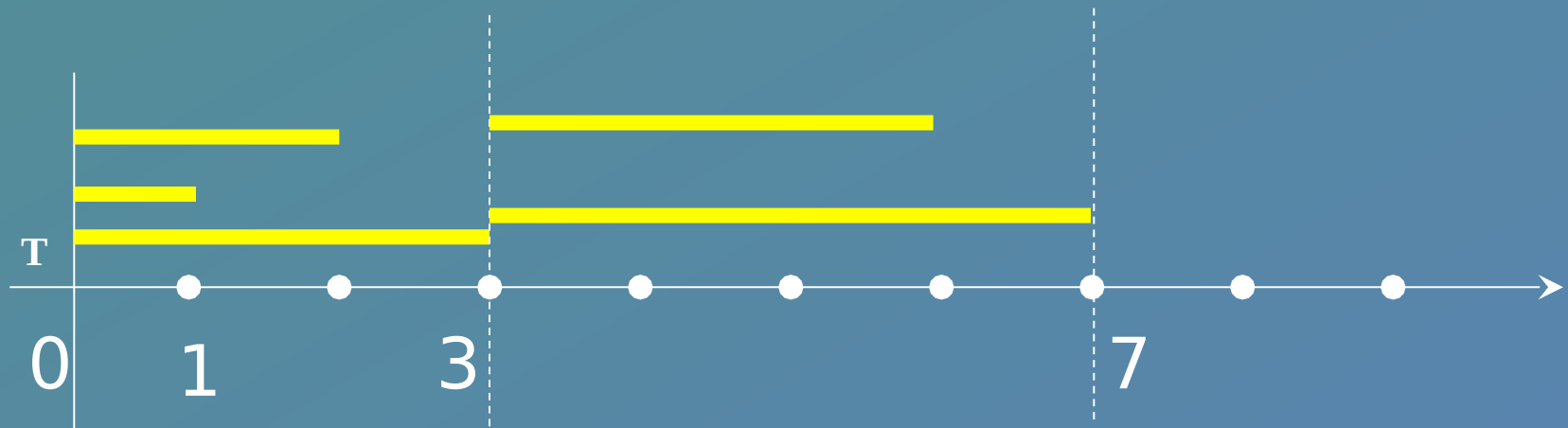
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T



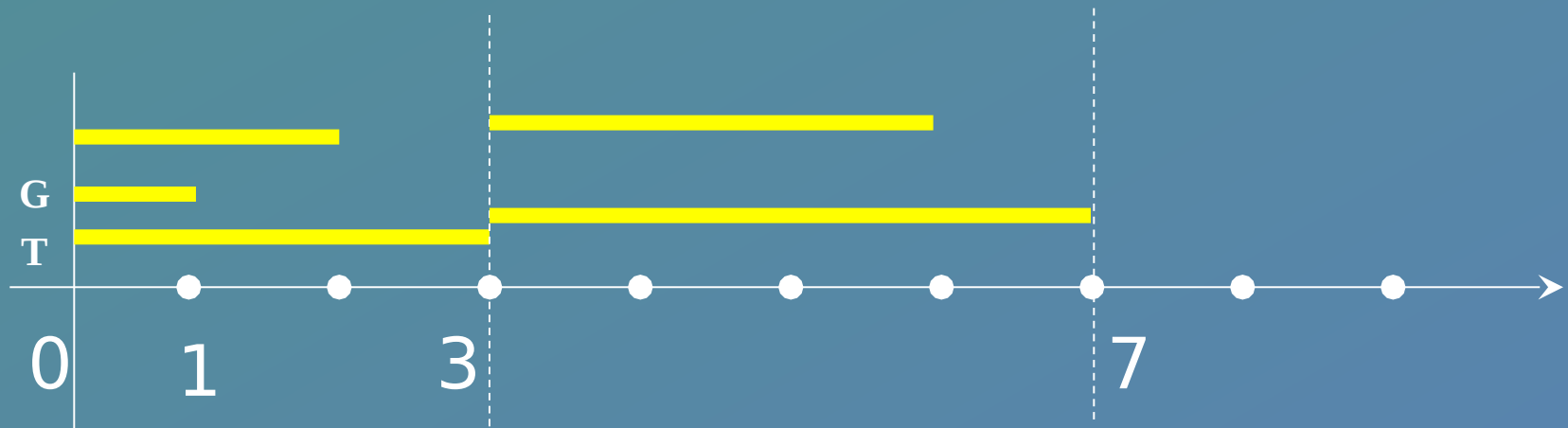
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



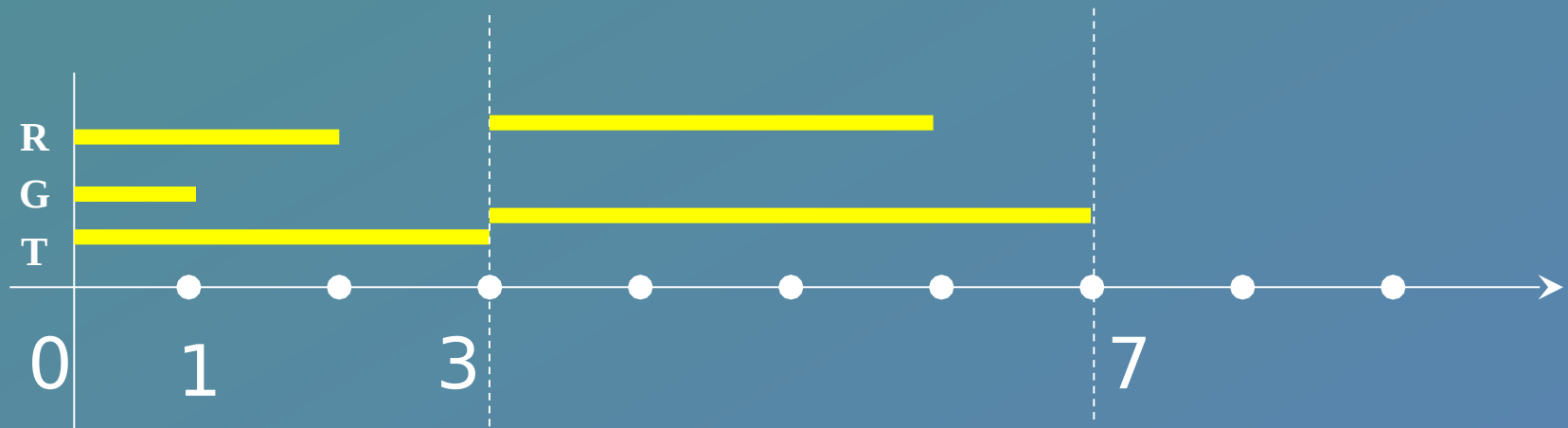
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



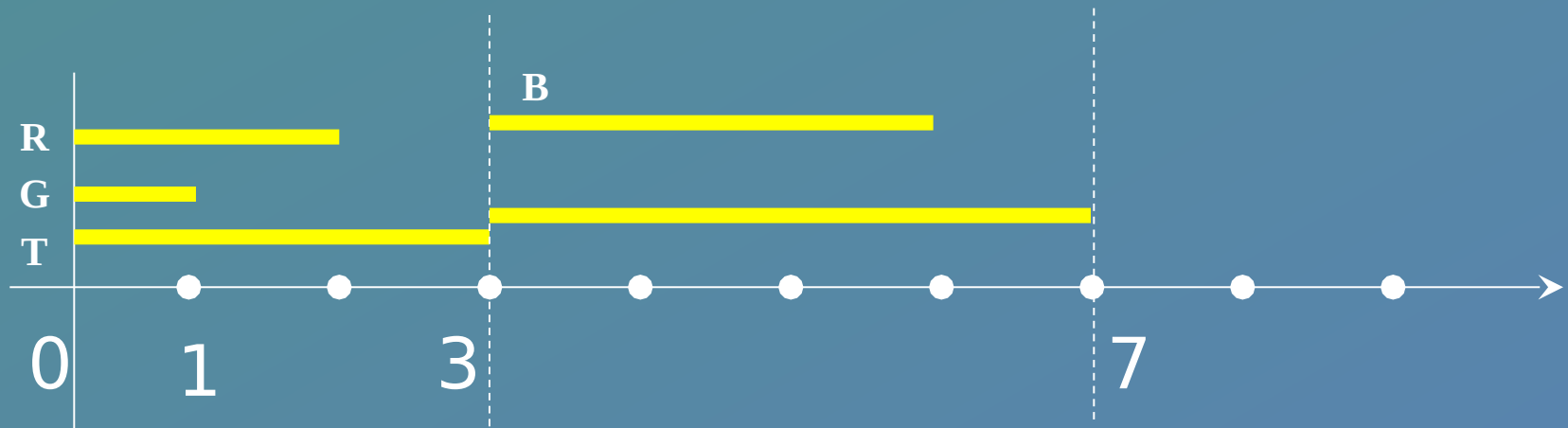
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



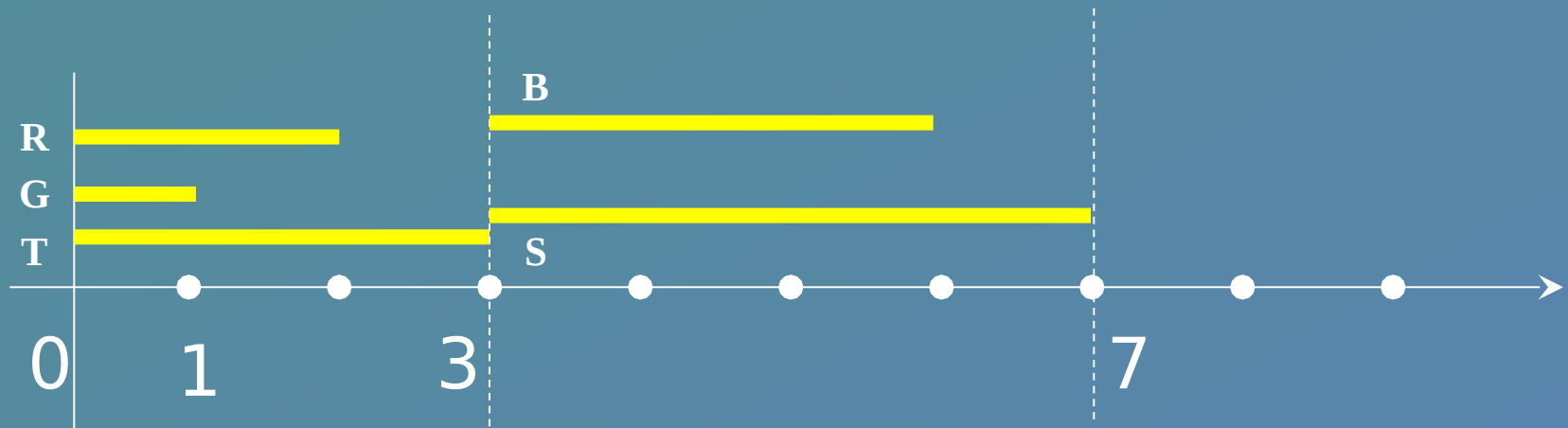
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



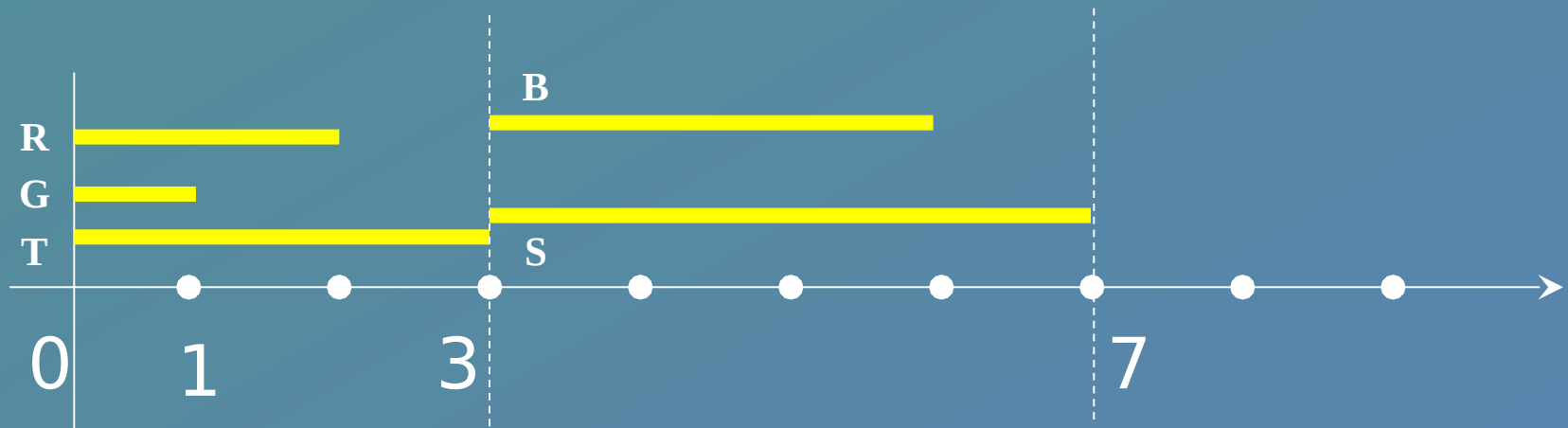
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



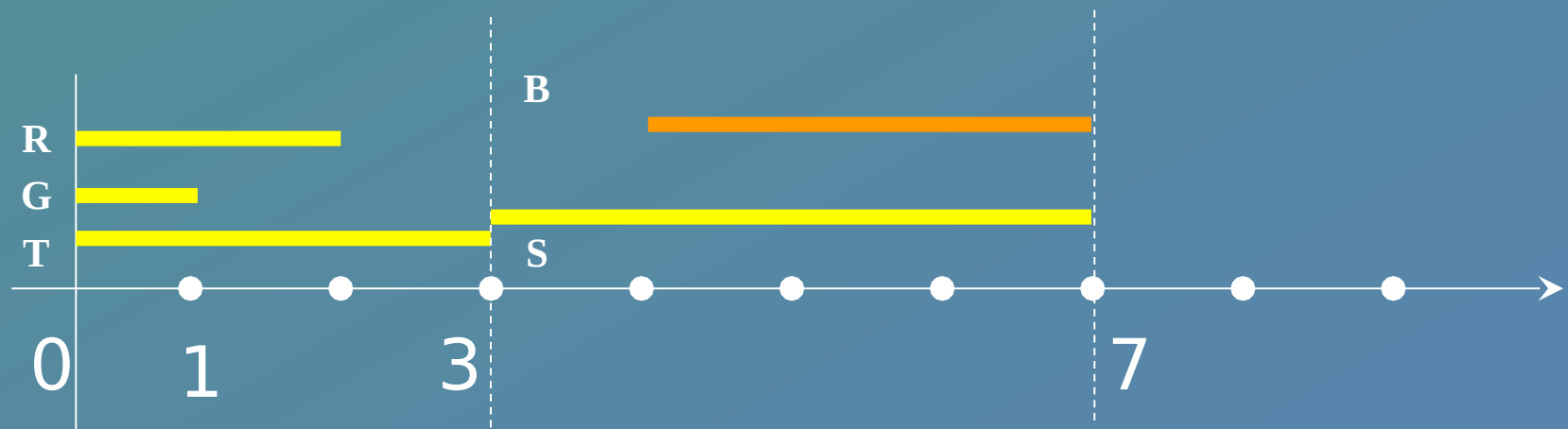
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



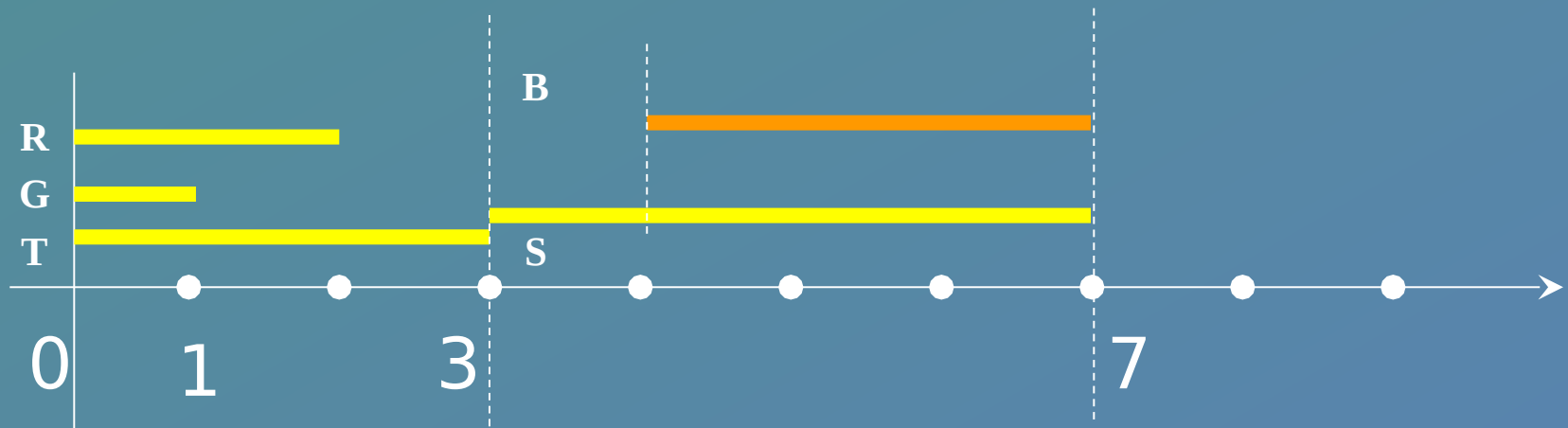
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



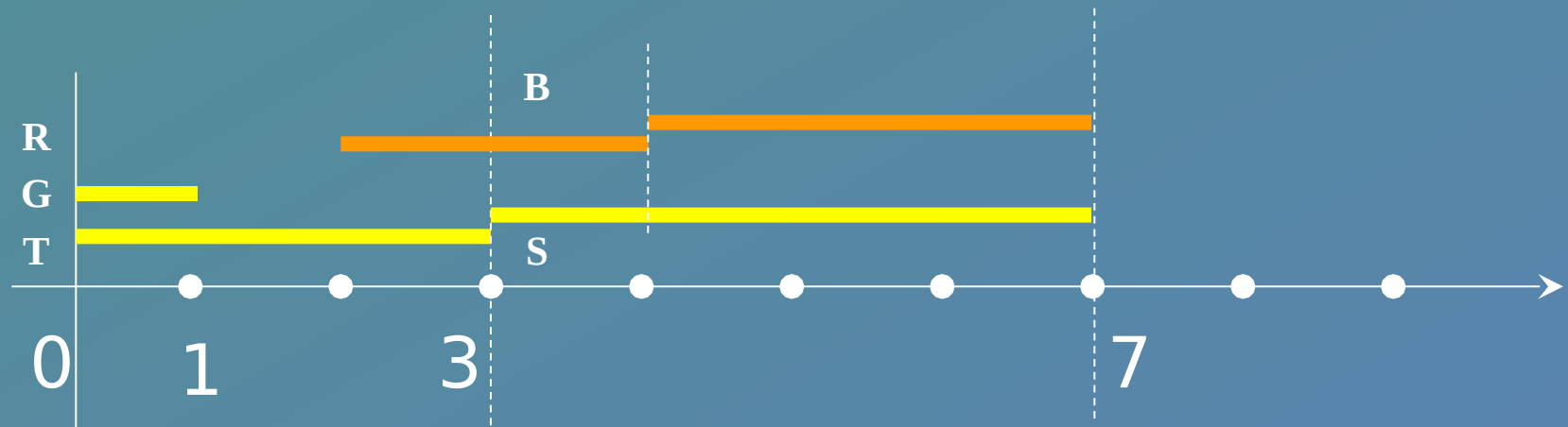
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



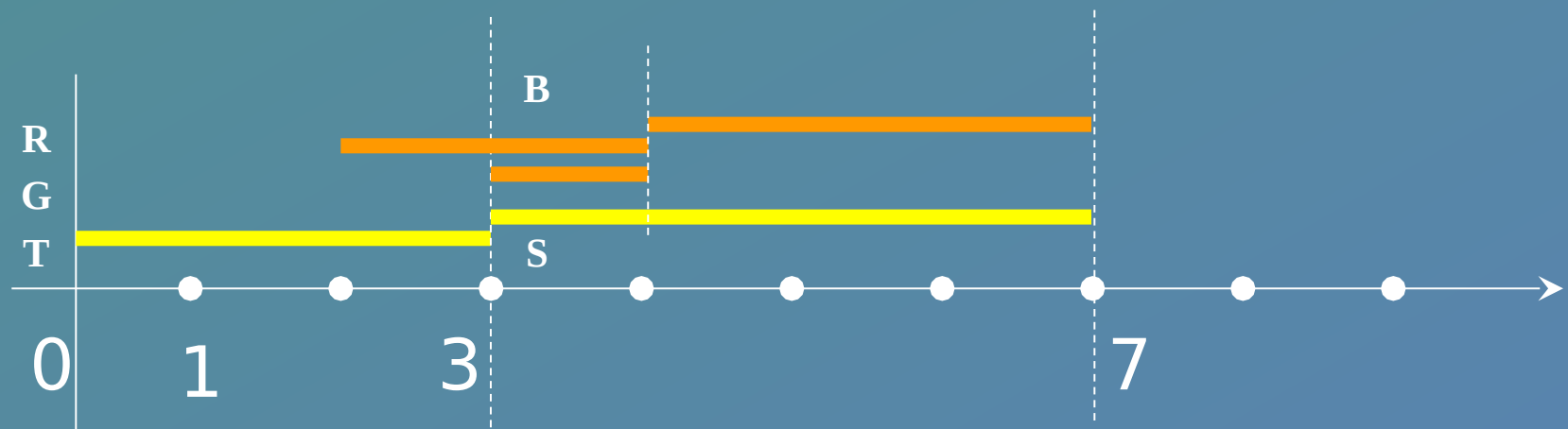
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



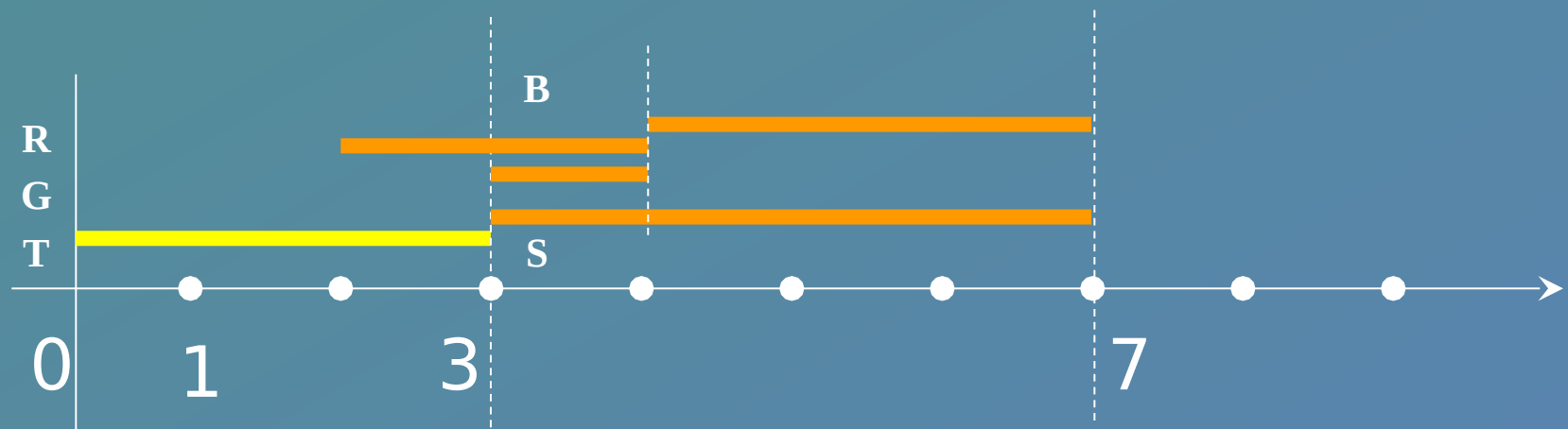
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



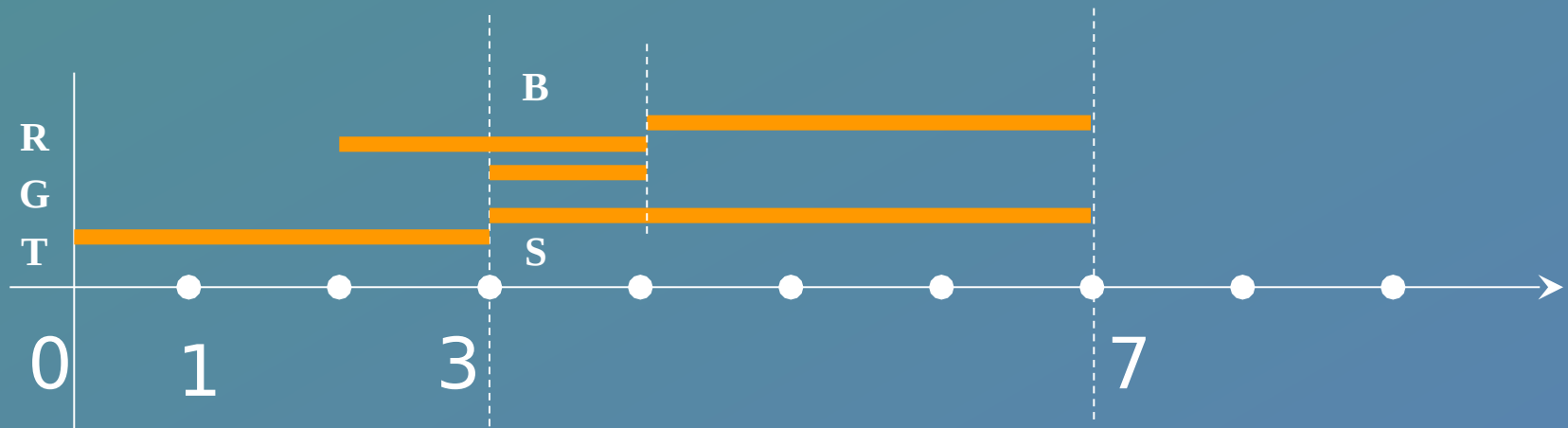
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



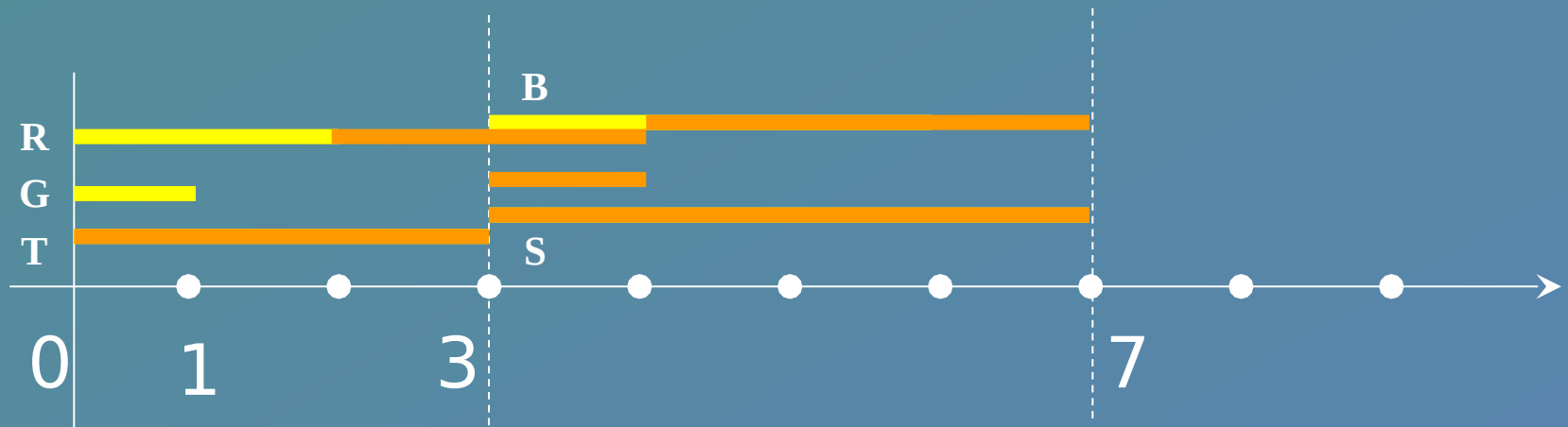
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



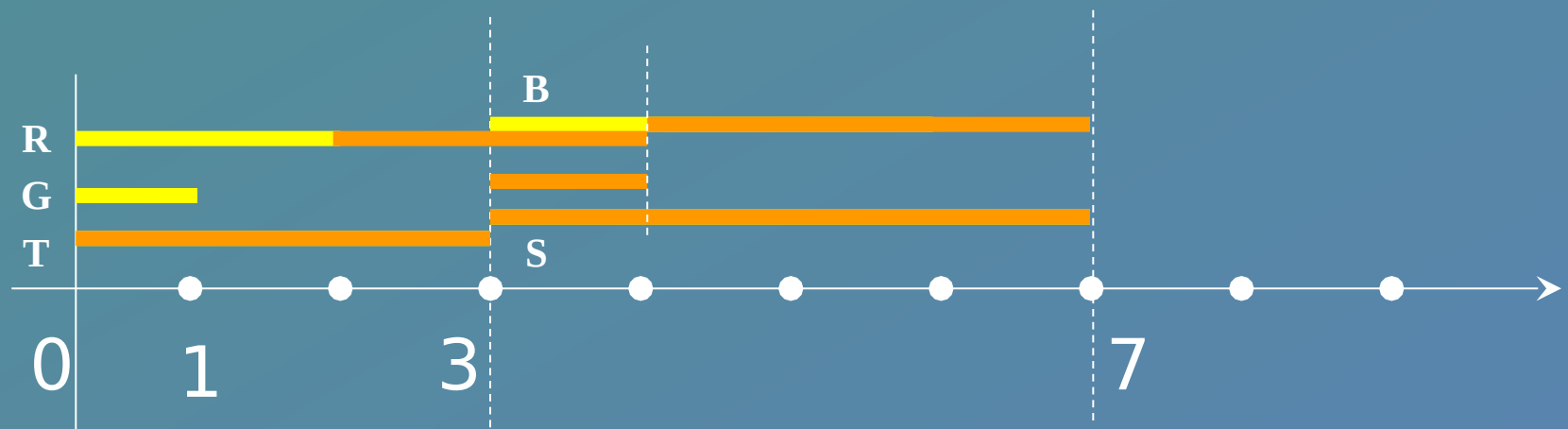
tâche	durée (jours)	contraintes ; après :	la date au plus tôt
T	3	—	0
G	1	—	0
R	2	—	0
B	3	T, G, R	3
S	4	T	3



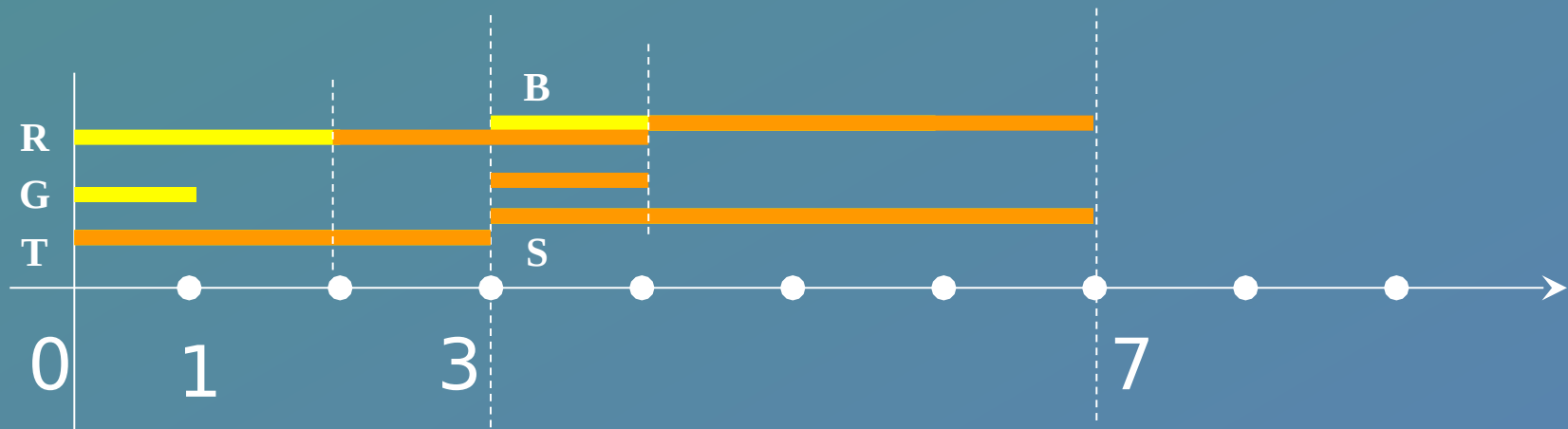
tâche	durée (jours)	contraintes ; après :	la date au plus tôt	la date au plus tard
T	3	—	0	0
G	1	—	0	3
R	2	—	0	2
B	3	T, G, R	3	4
S	4	T	3	3



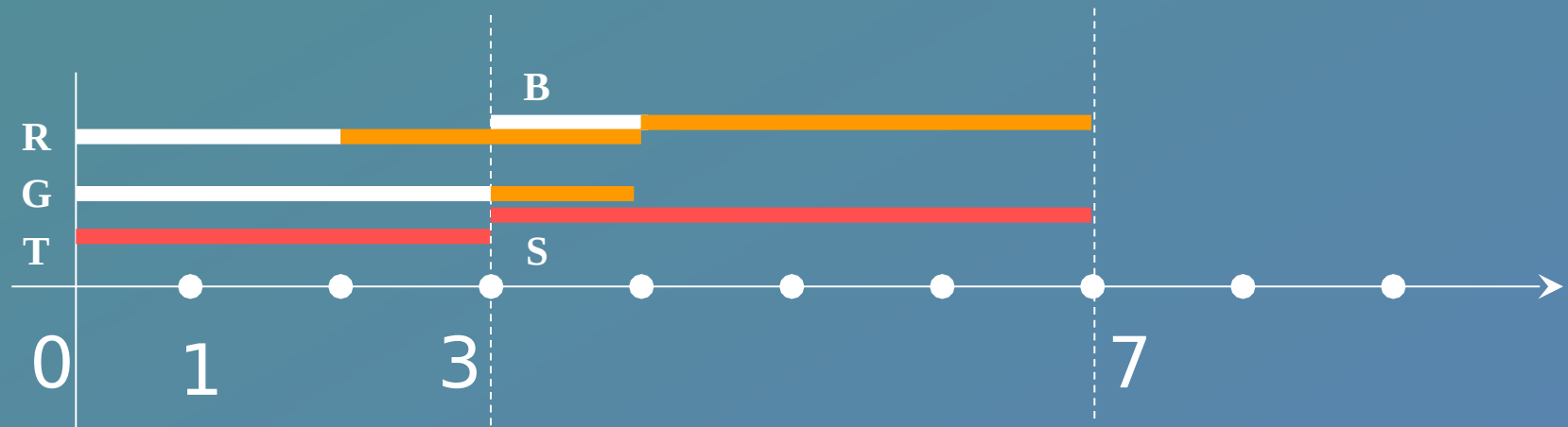
tâche	durée (jours)	contraintes ; après :	la date au plus tôt	la date au plus tard
T	3	—	0	0
G	1	—	0	3
R	2	—	0	2
B	3	T, G, R	3	4
S	4	T	3	3



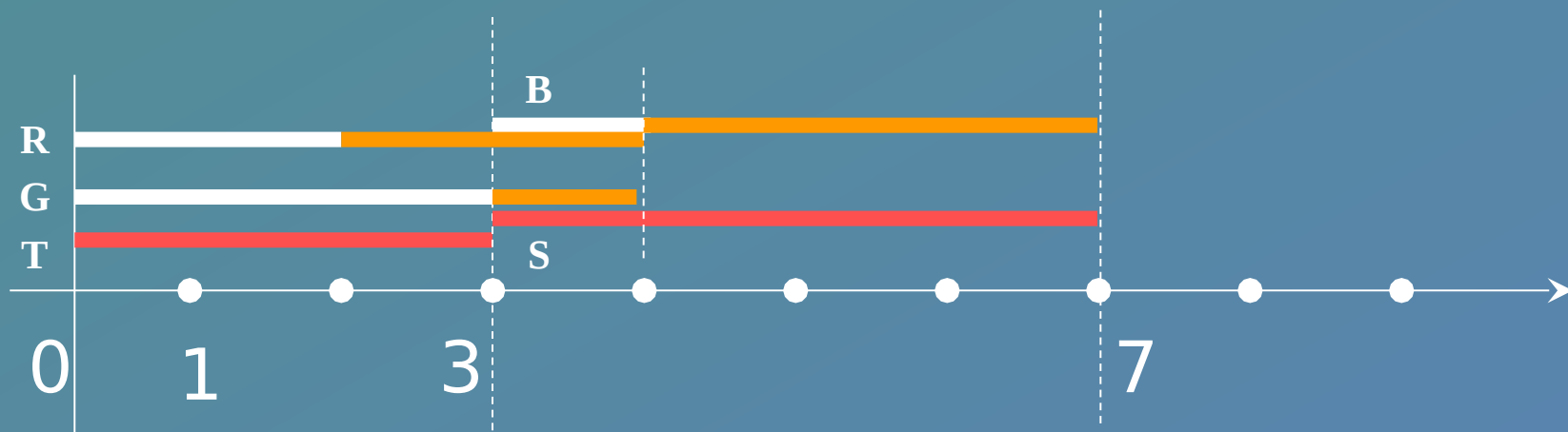
tâche	durée (jours)	contraintes ; après :	la date au plus tôt	la date au plus tard
T	3	—	0	0
G	1	—	0	3
R	2	—	0	2
B	3	T, G, R	3	4
S	4	T	3	3



tâche	durée (jours)	contraintes ; après :	la date au plus tôt	la date au plus tard	la marge
T	3	—	0	0	0
G	1	—	0	3	3
R	2	—	0	2	2
B	3	T, G, R	3	4	1
S	4	T	3	3	0



tâche	durée (jours)	contraintes ; après :	la date au plus tôt	la date au plus tard	la marge
T	3	—	0	0	0
G	1	—	0	3	3
R	2	—	0	2	2
B	3	T, G, R	3	4	1
S	4	T	3	3	0





potentiels-tâches



potentiels-tâches

T



potentiels-tâches

T

G



potentiels-tâches

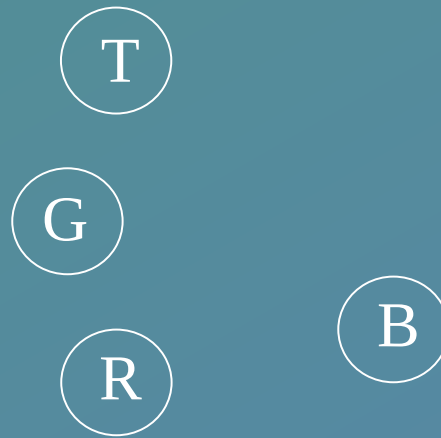
T

G

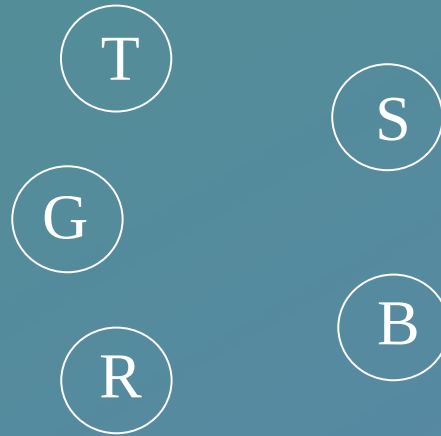
R



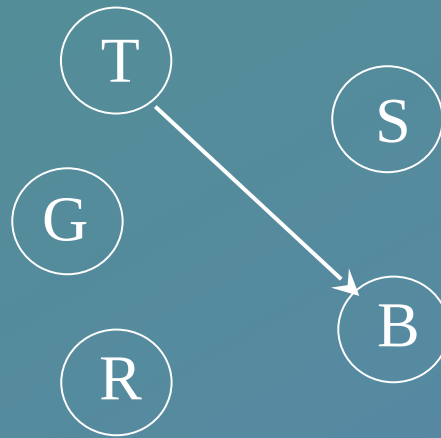
potentiels-tâches



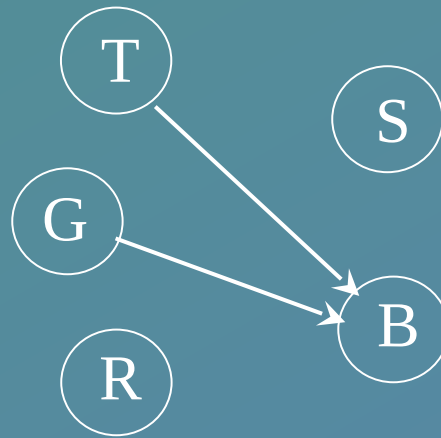
potentiels-tâches



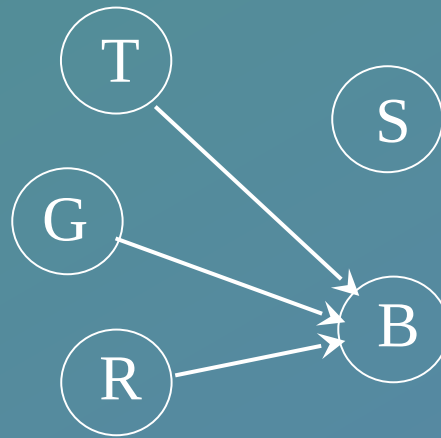
potentiels-tâches



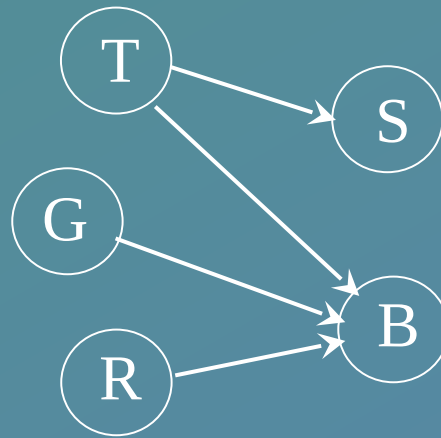
potentiels-tâches



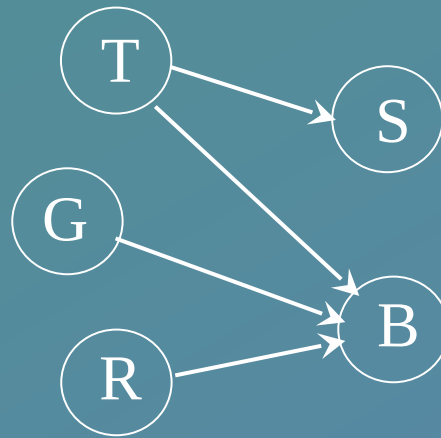
potentiels-tâches



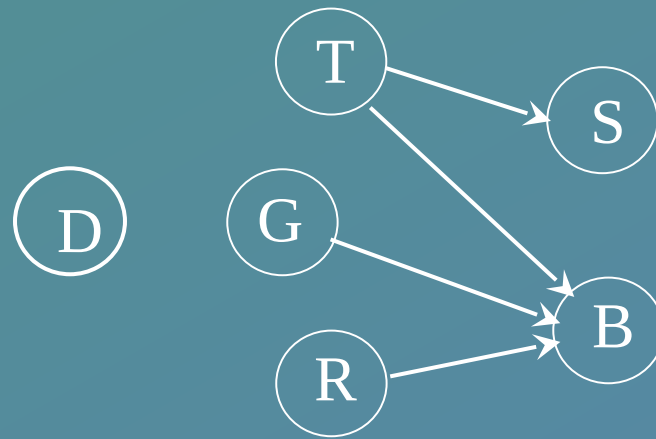
potentiels-tâches



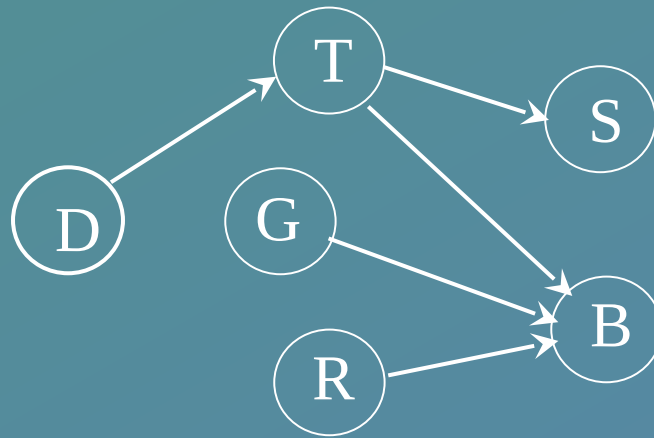
potentiels-tâches



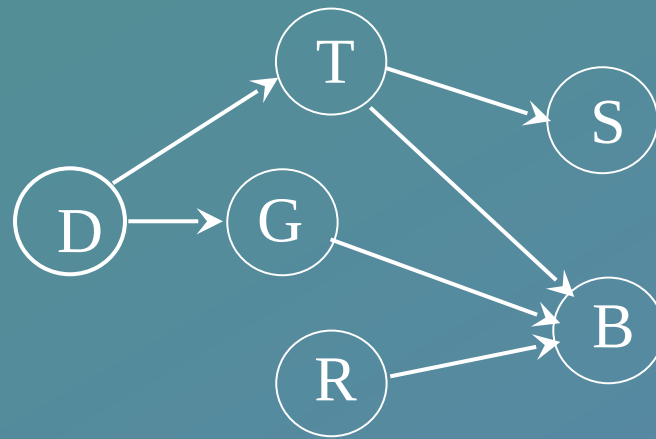
potentiels-tâches



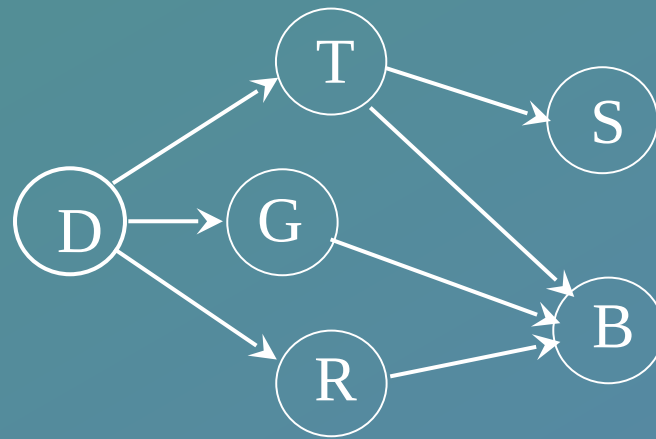
potentiels-tâches



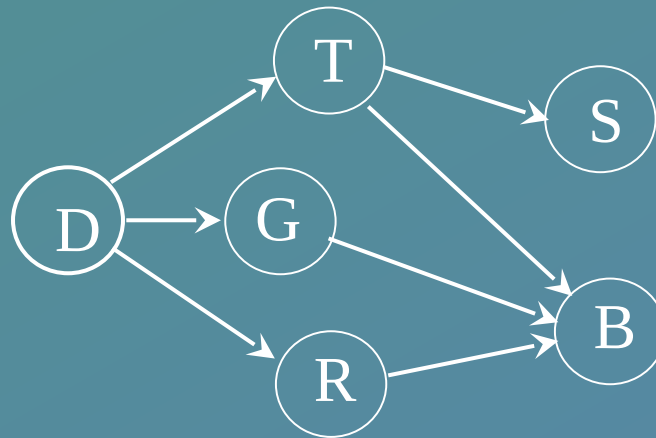
potentiels-tâches



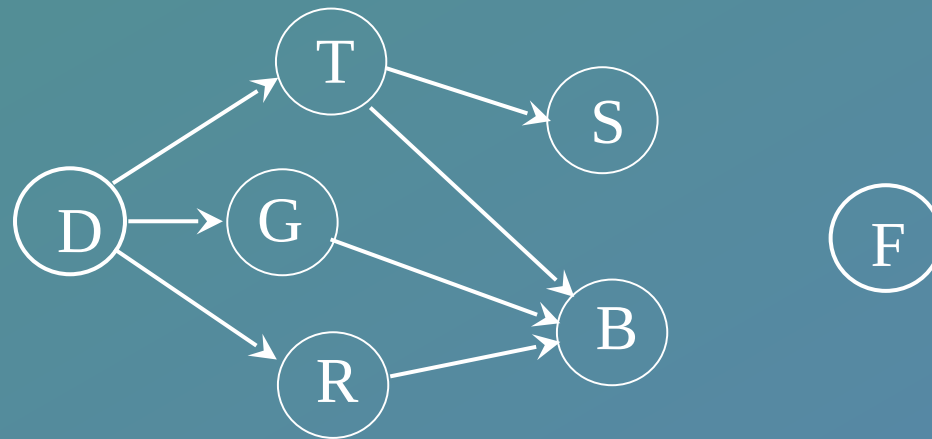
potentiels-tâches



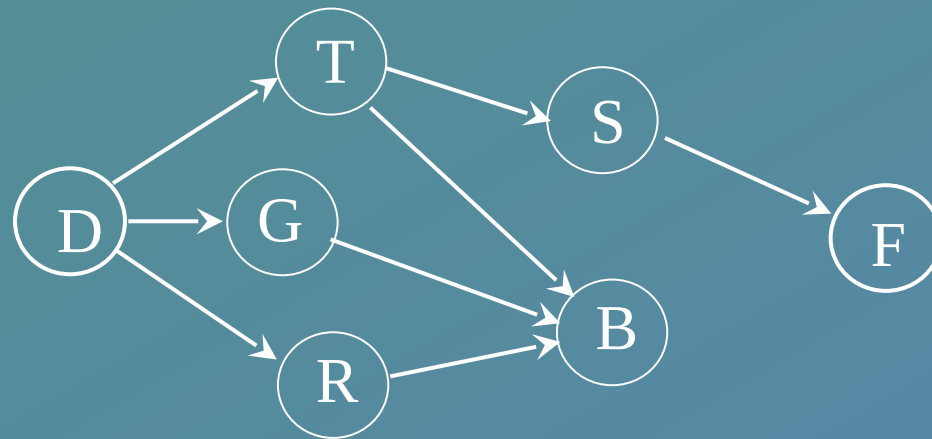
potentiels-tâches



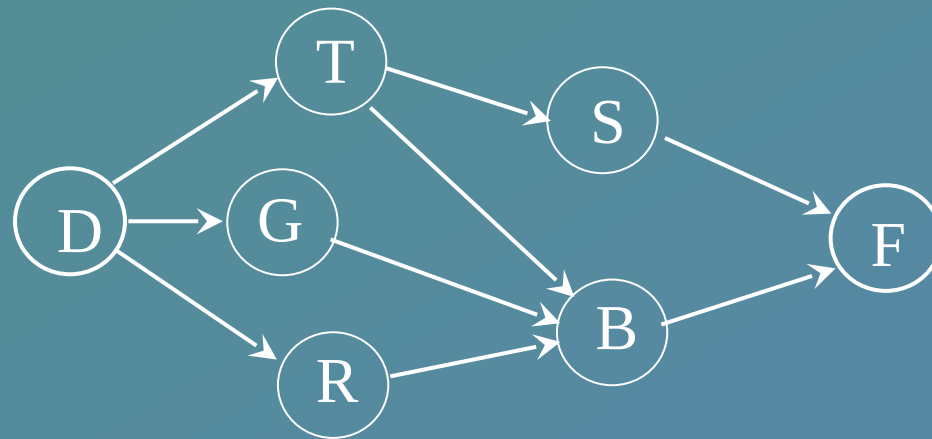
potentiels-tâches



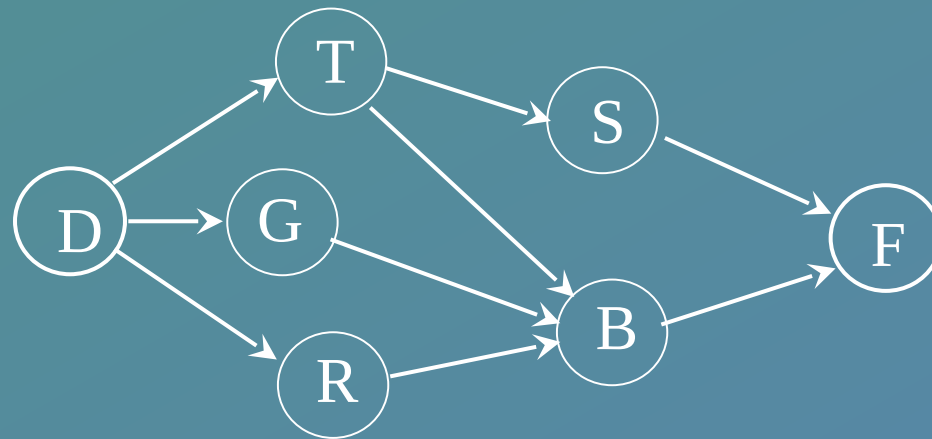
potentiels-tâches



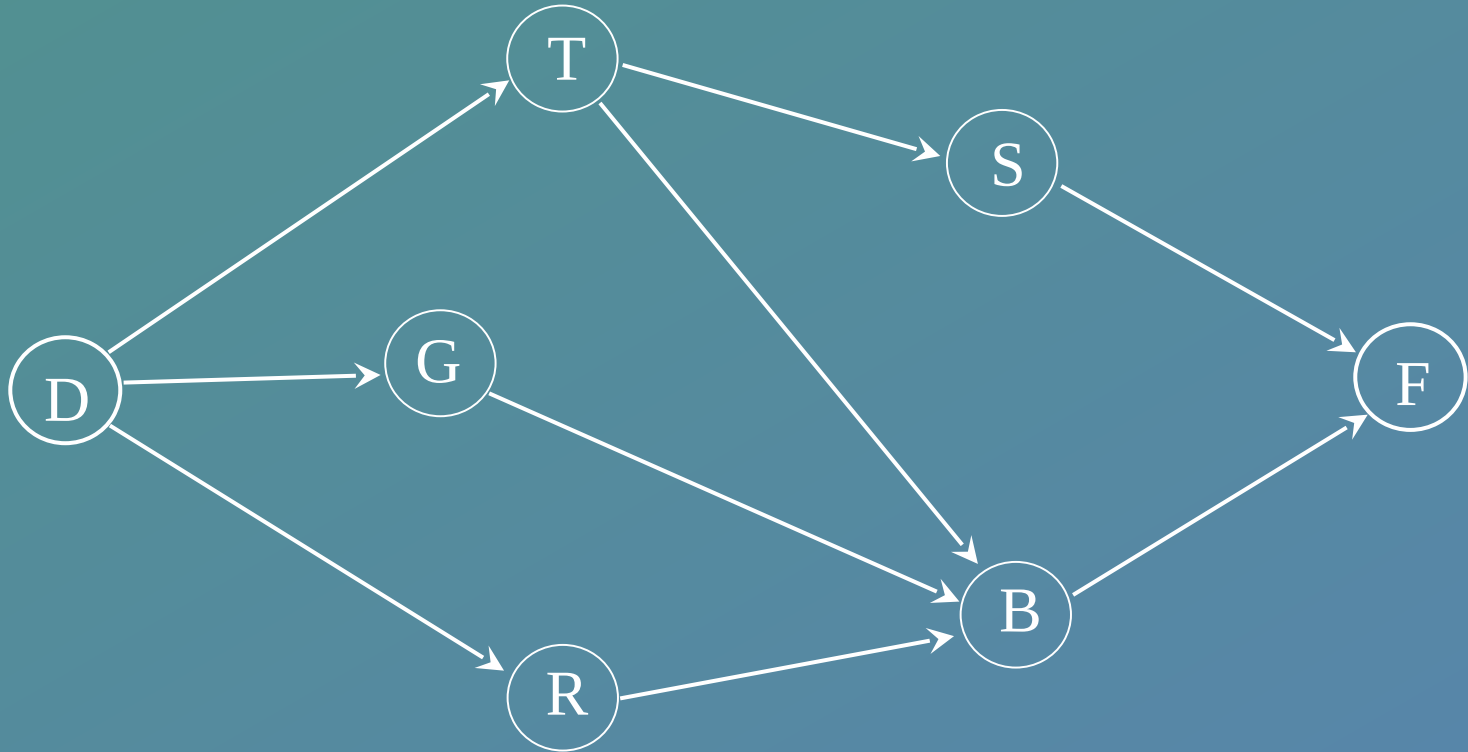
potentiels-tâches



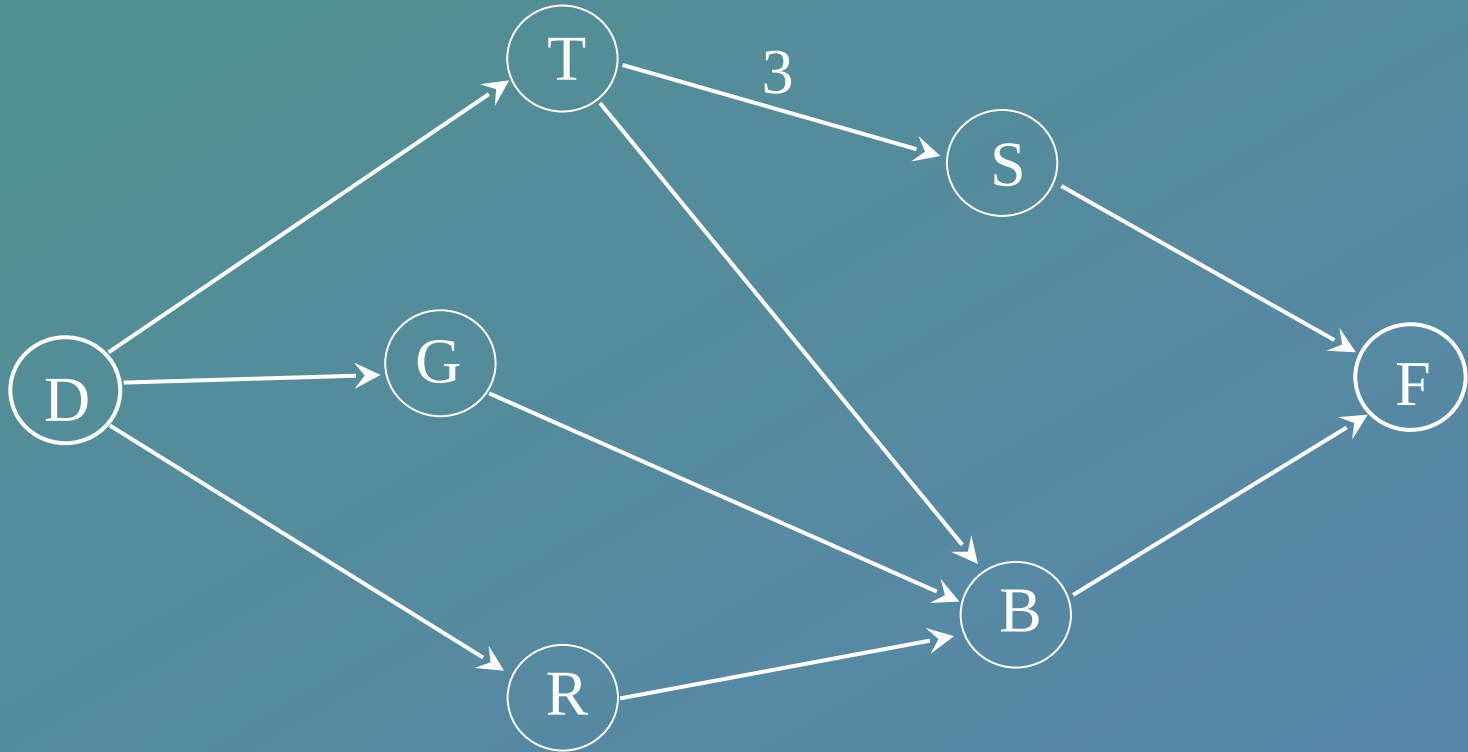
potentiels-tâches



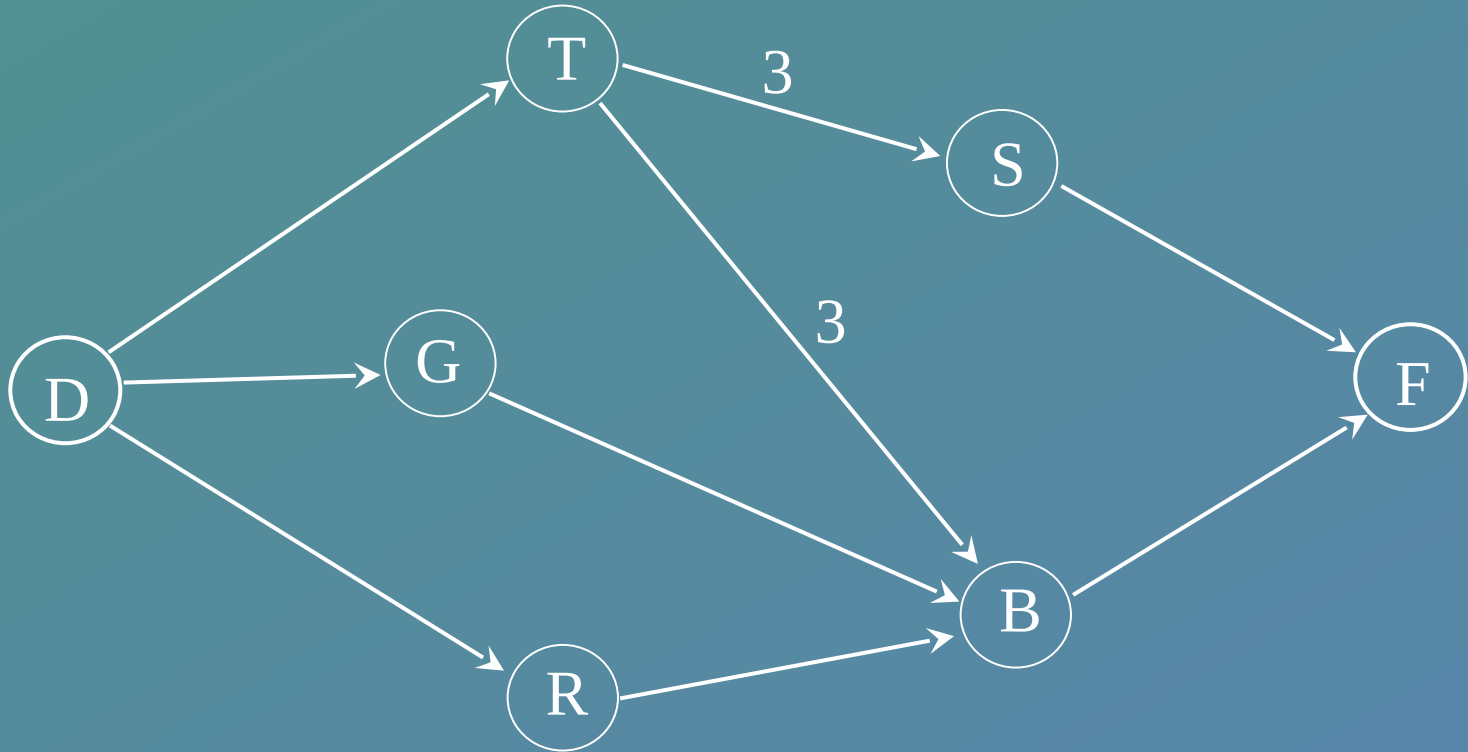
potentiels-tâches



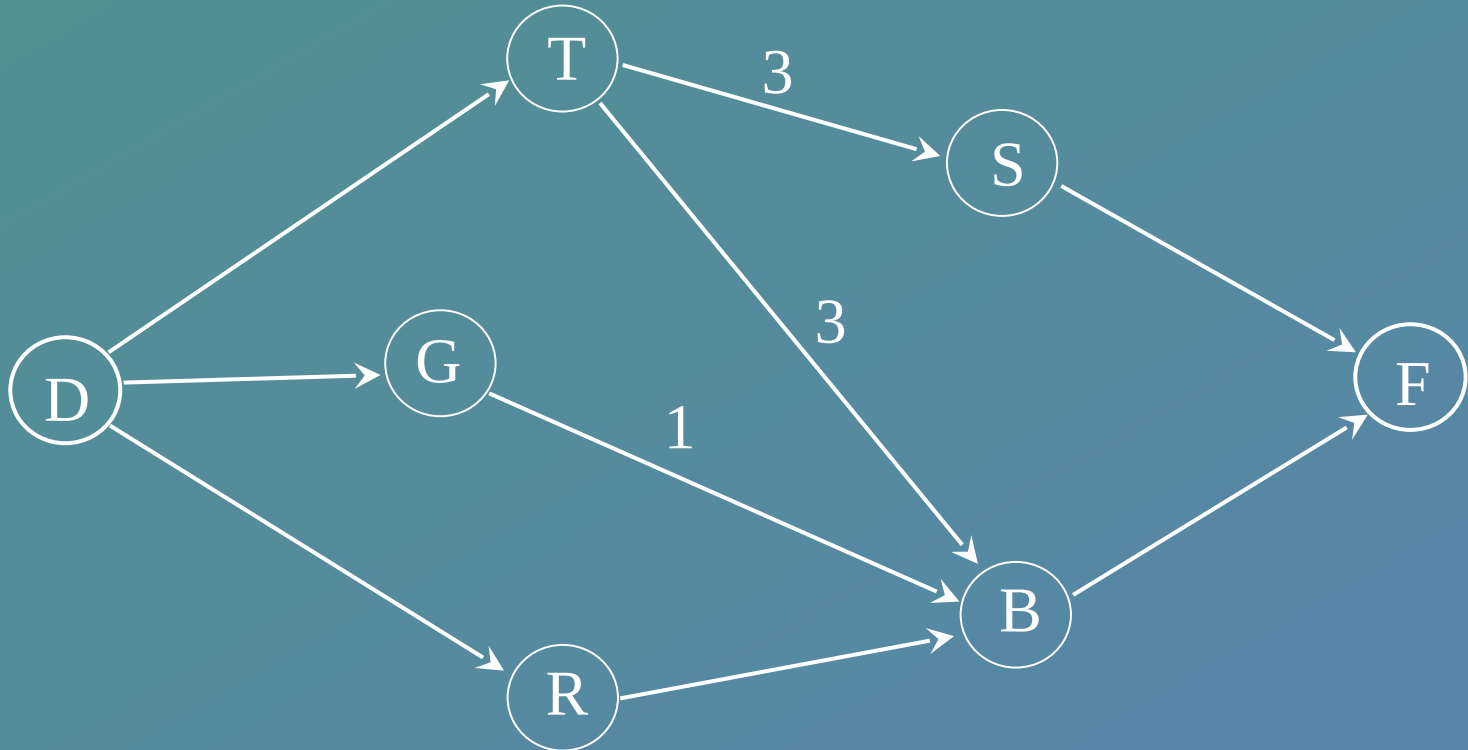
potentiels-tâches



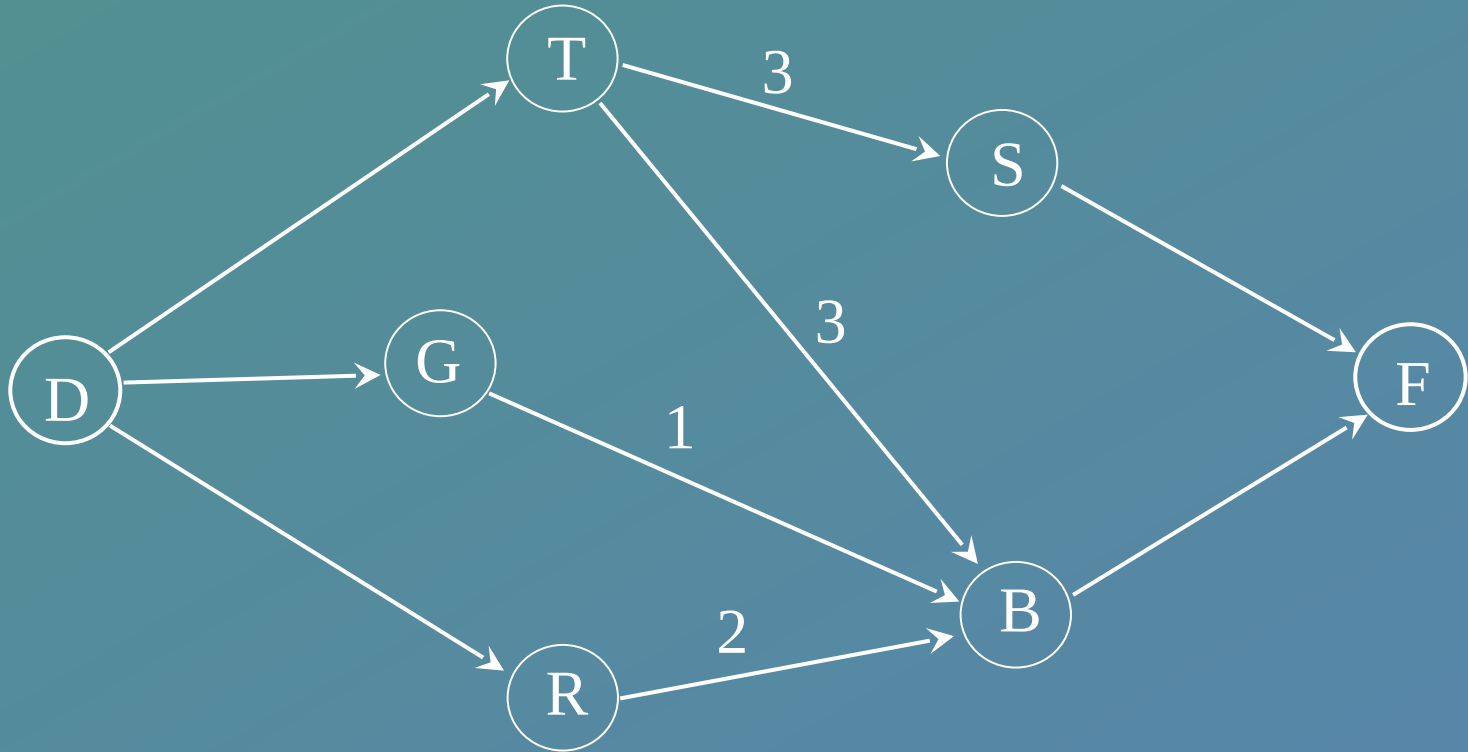
potentiels-tâches



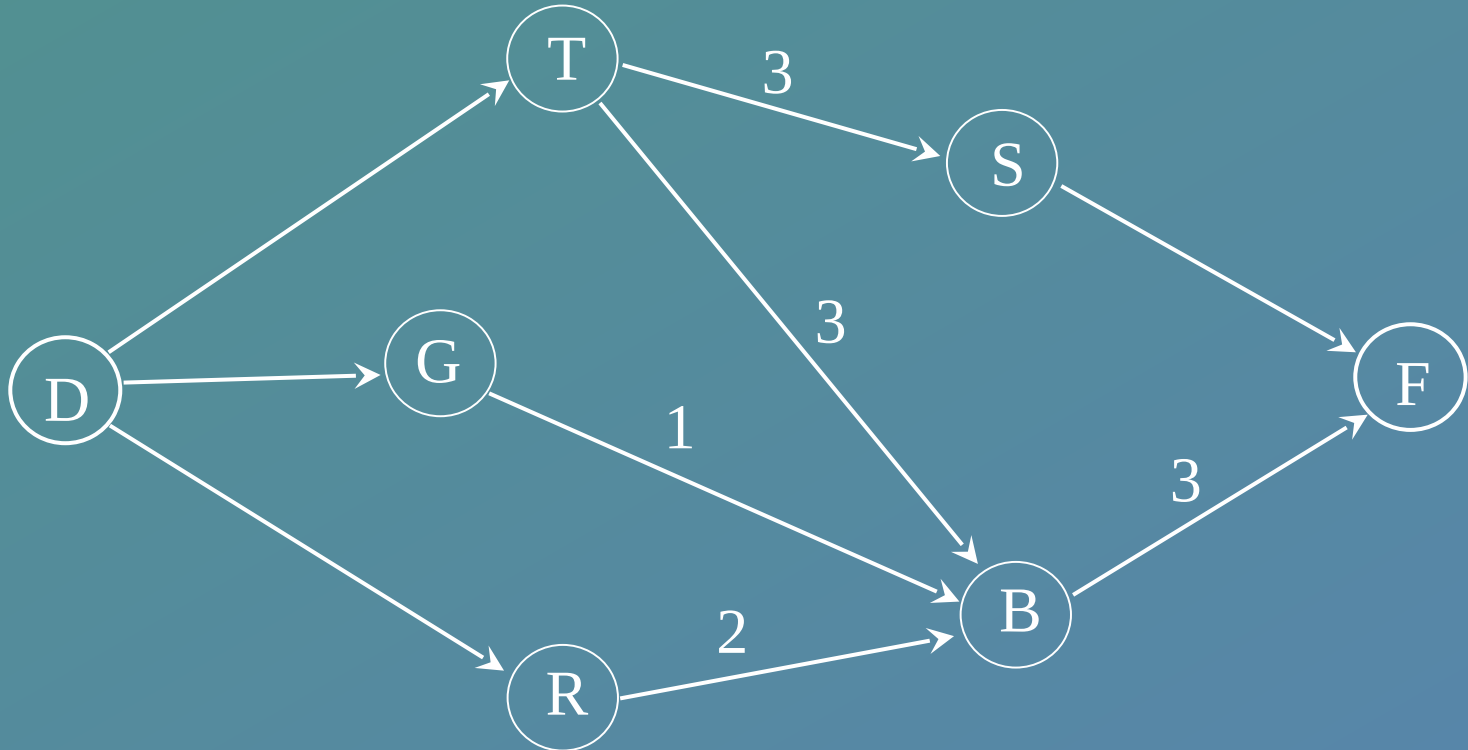
potentiels-tâches



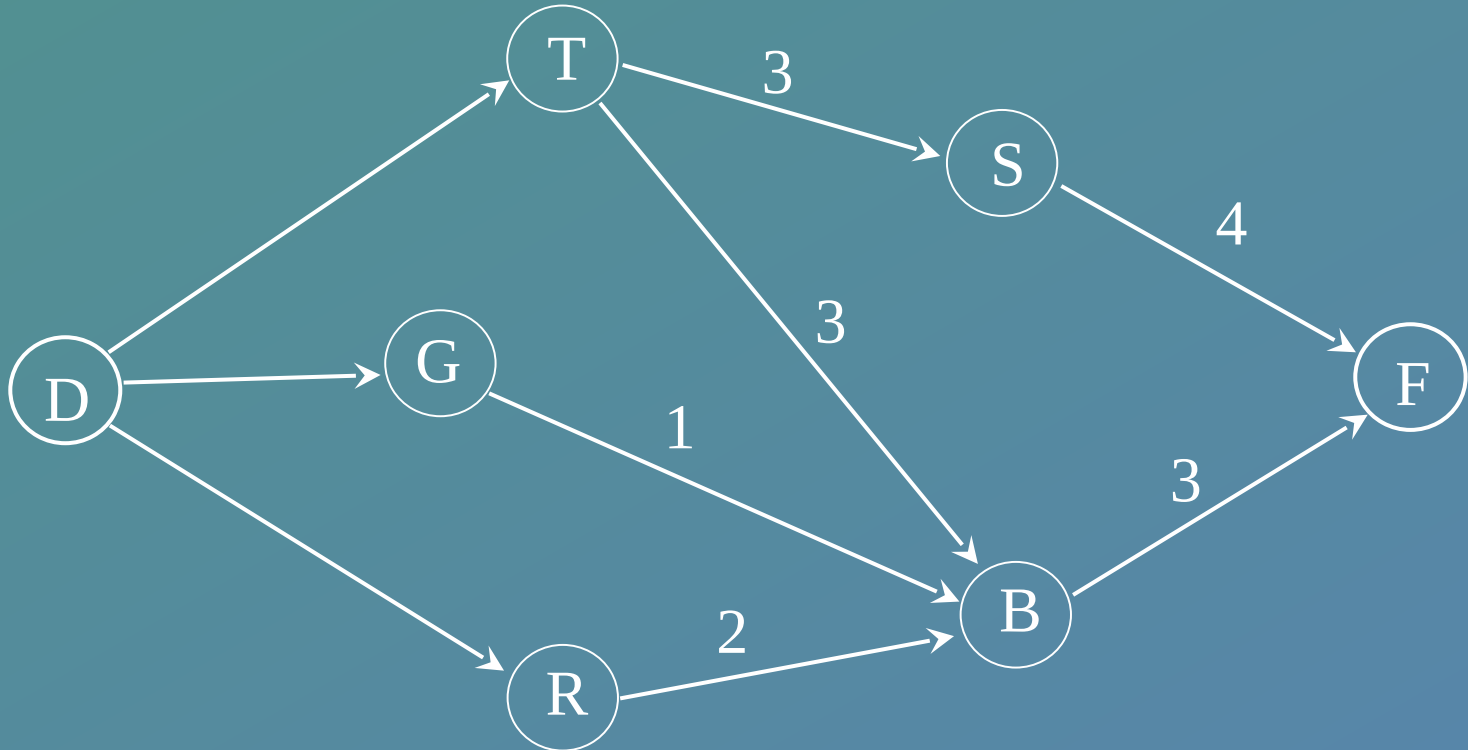
potentiels-tâches



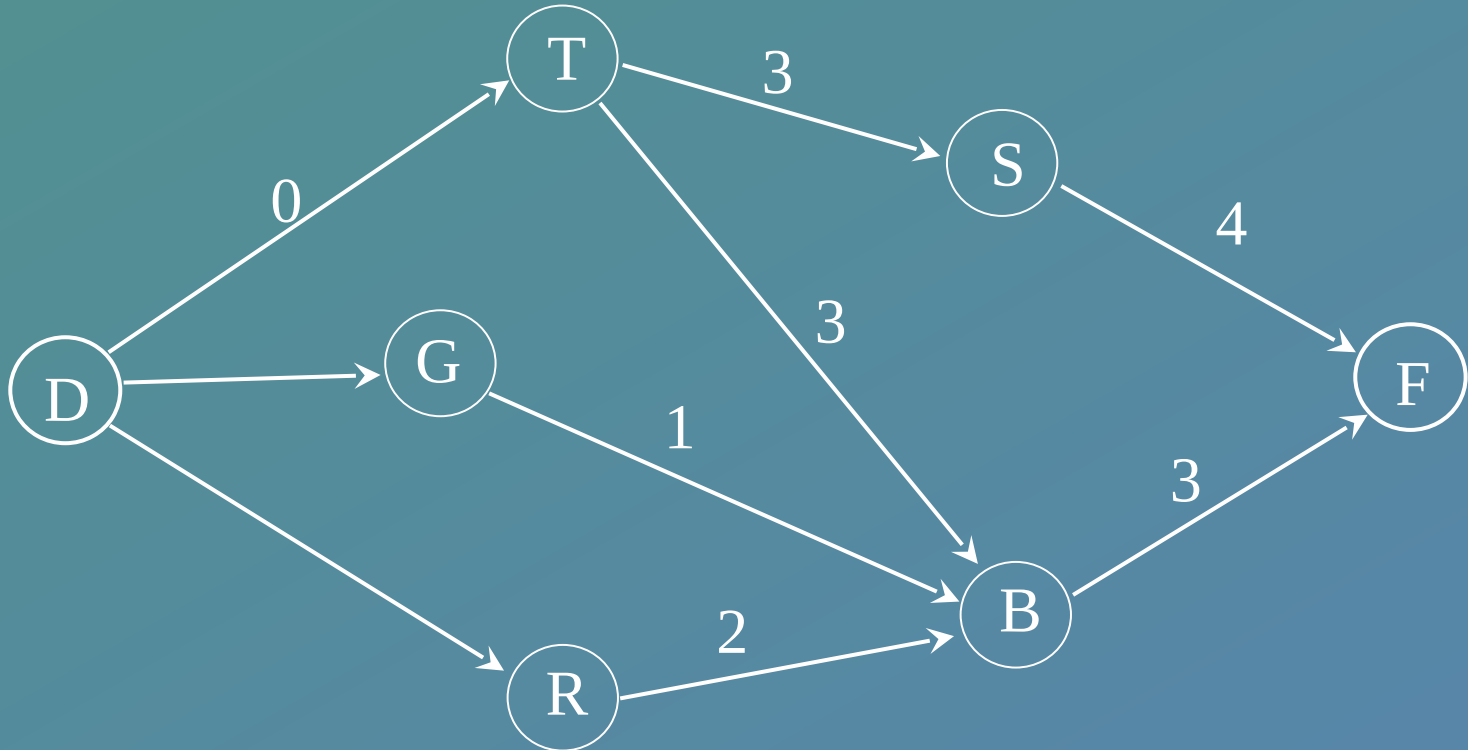
potentiels-tâches



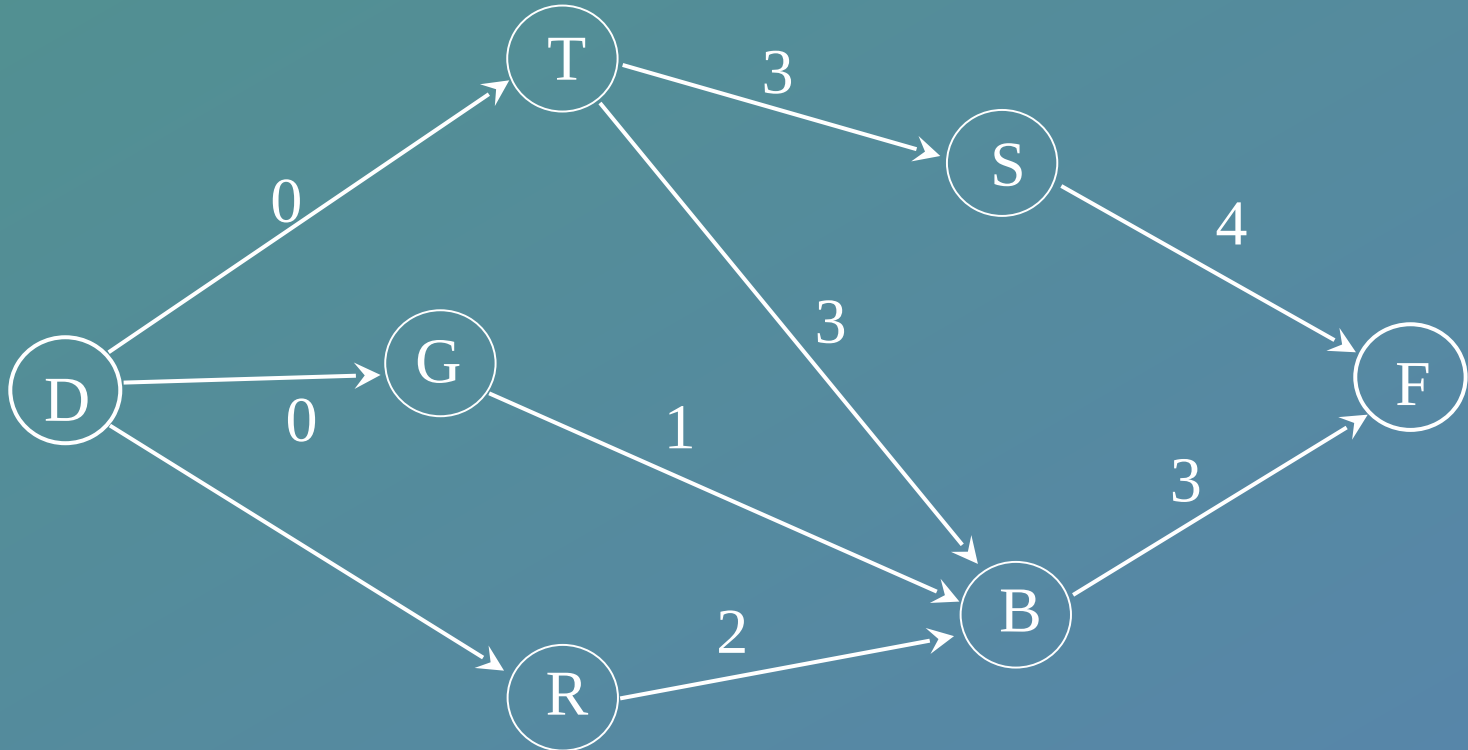
potentiels-tâches



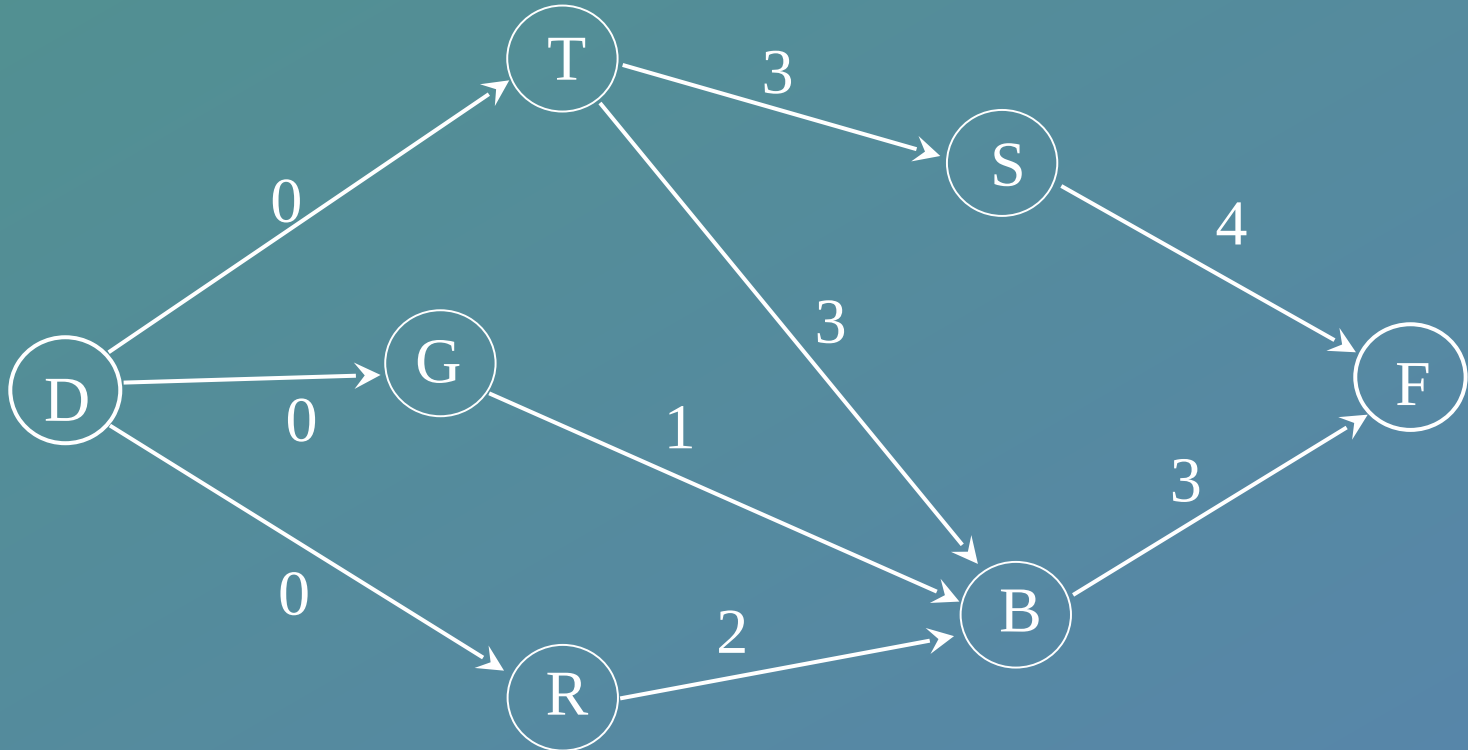
potentiels-tâches



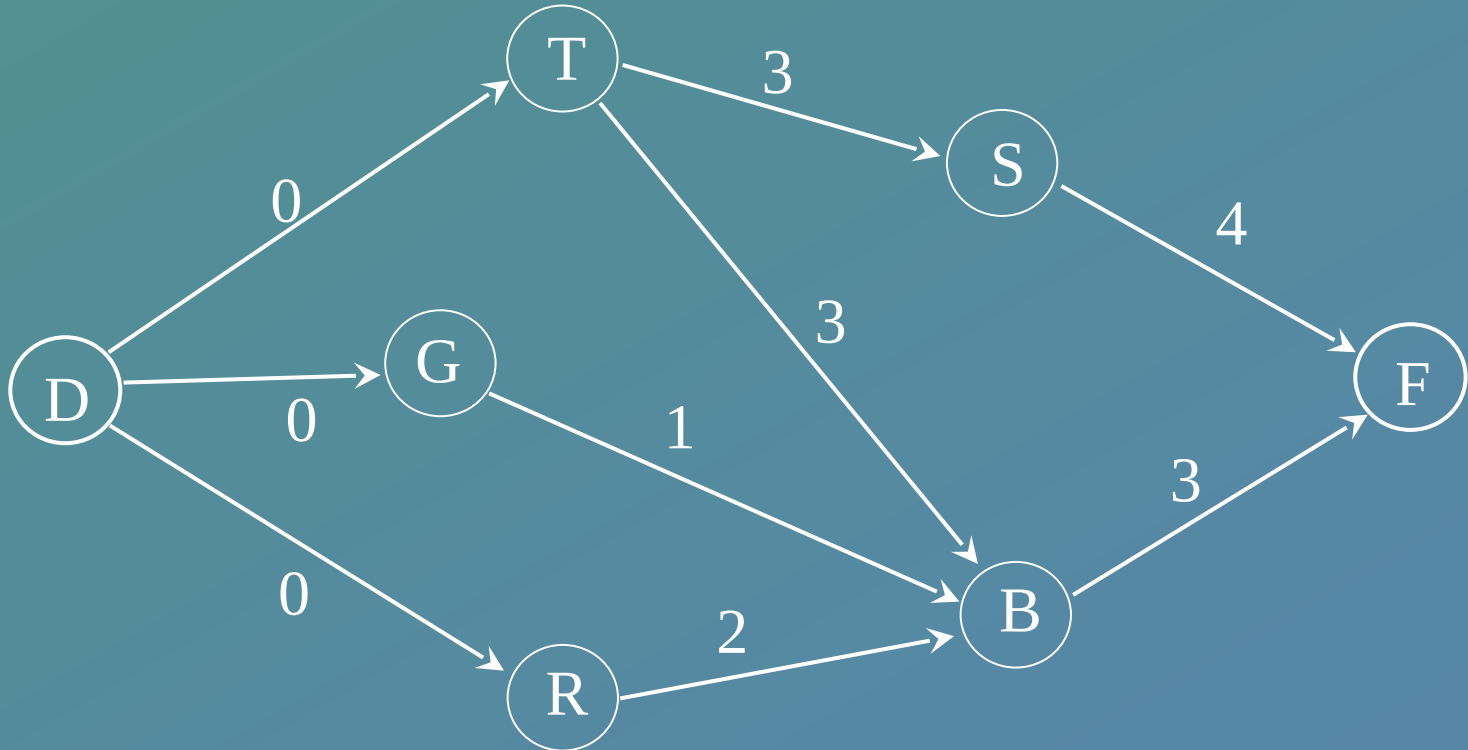
potentiels-tâches



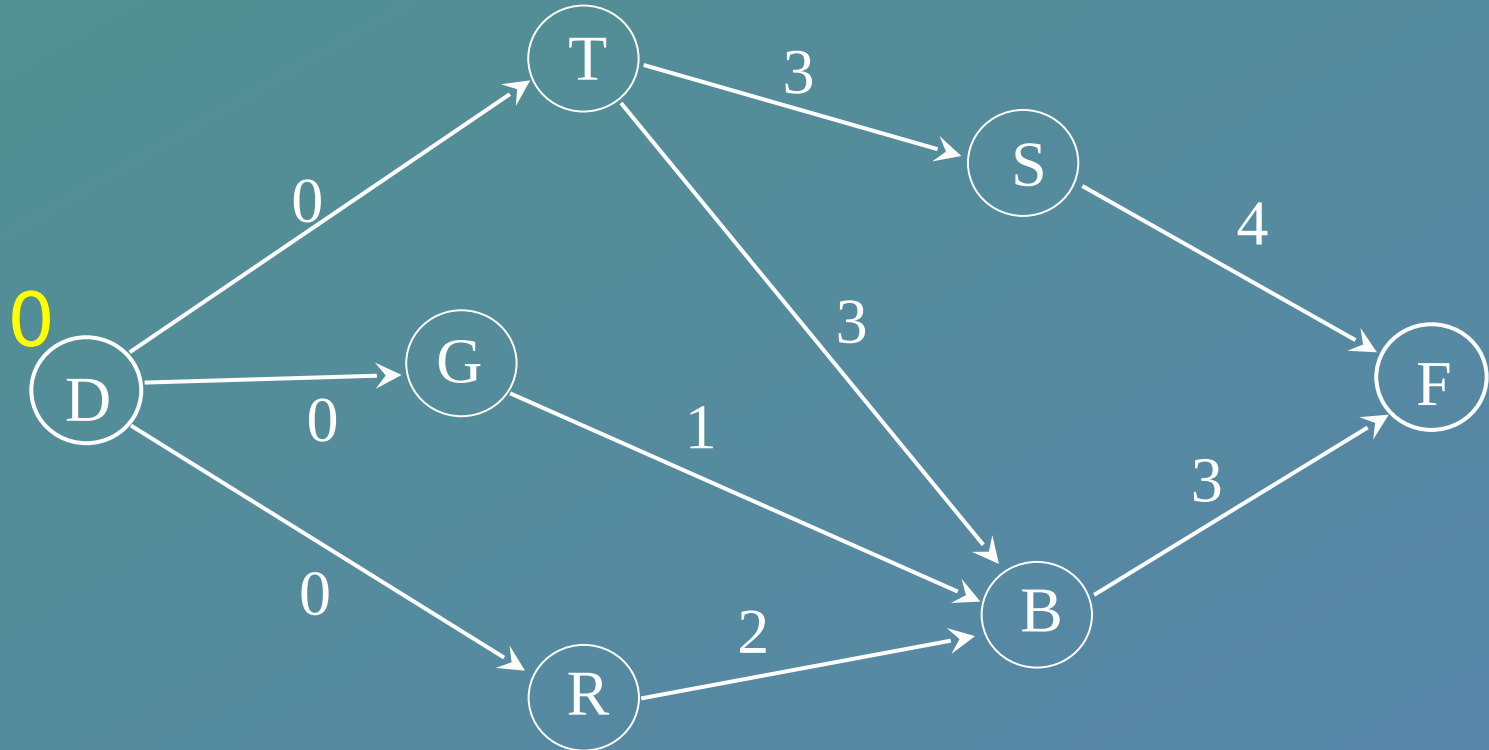
potentiels-tâches



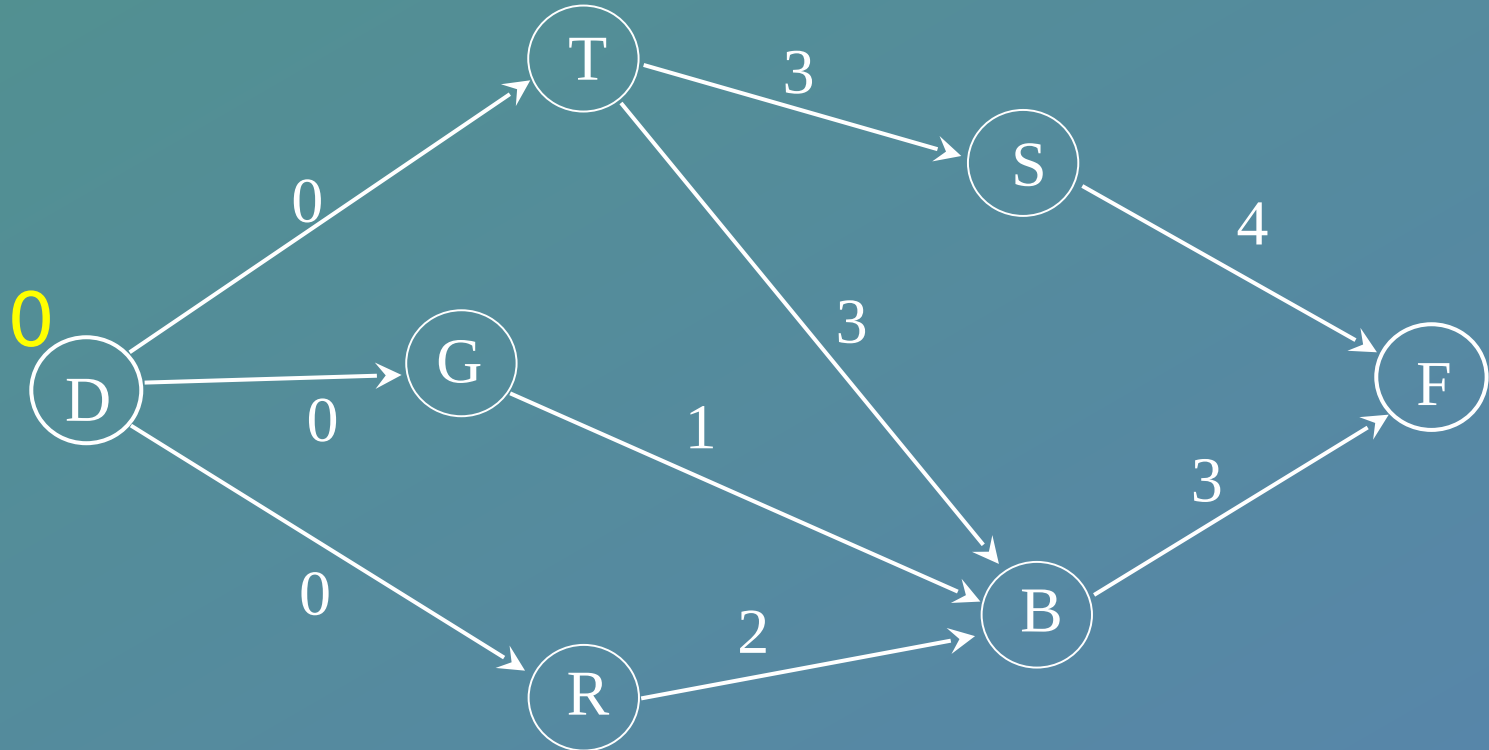
potentiels-tâches



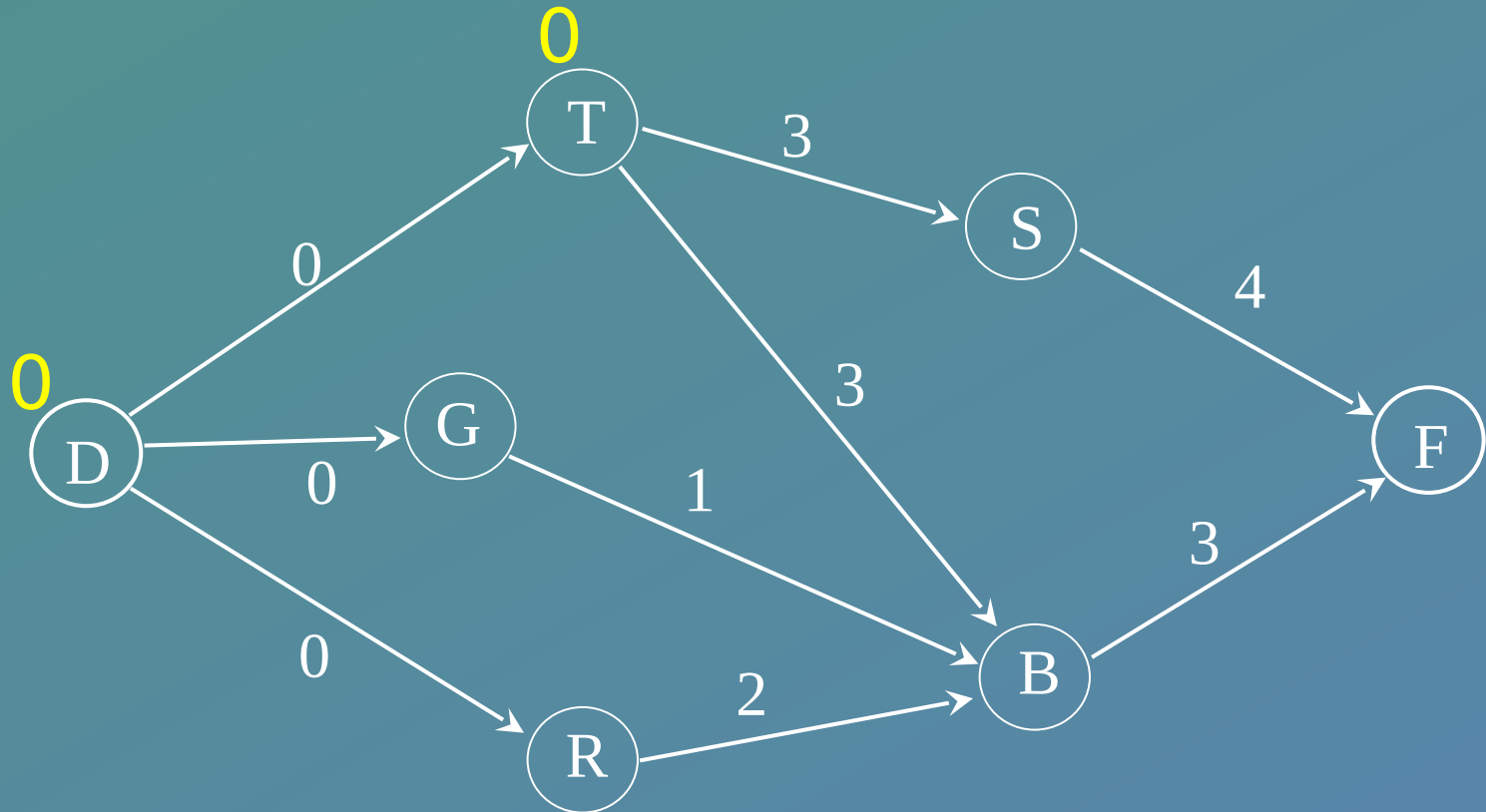
potentiels-tâches



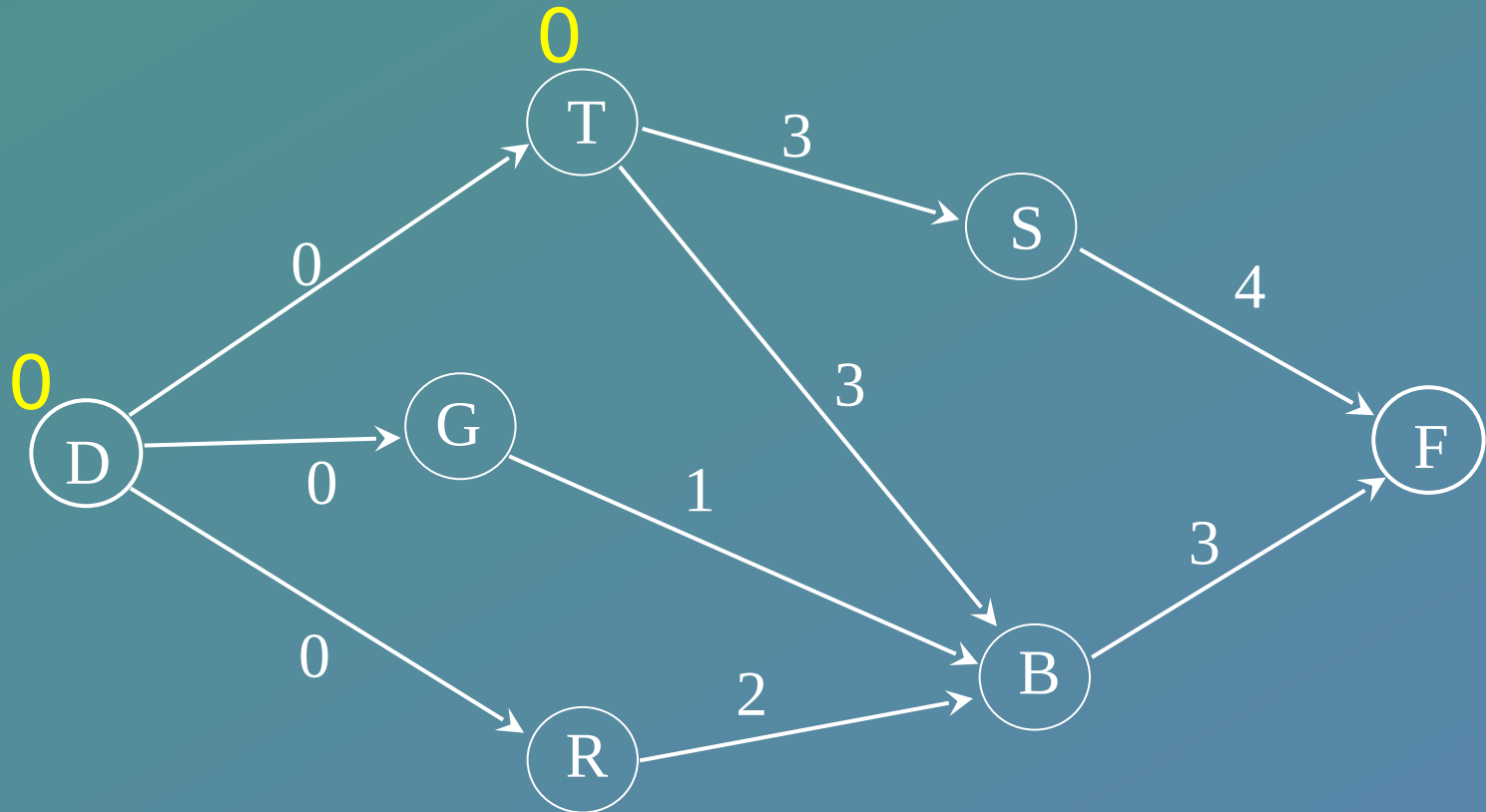
potentiels-tâches



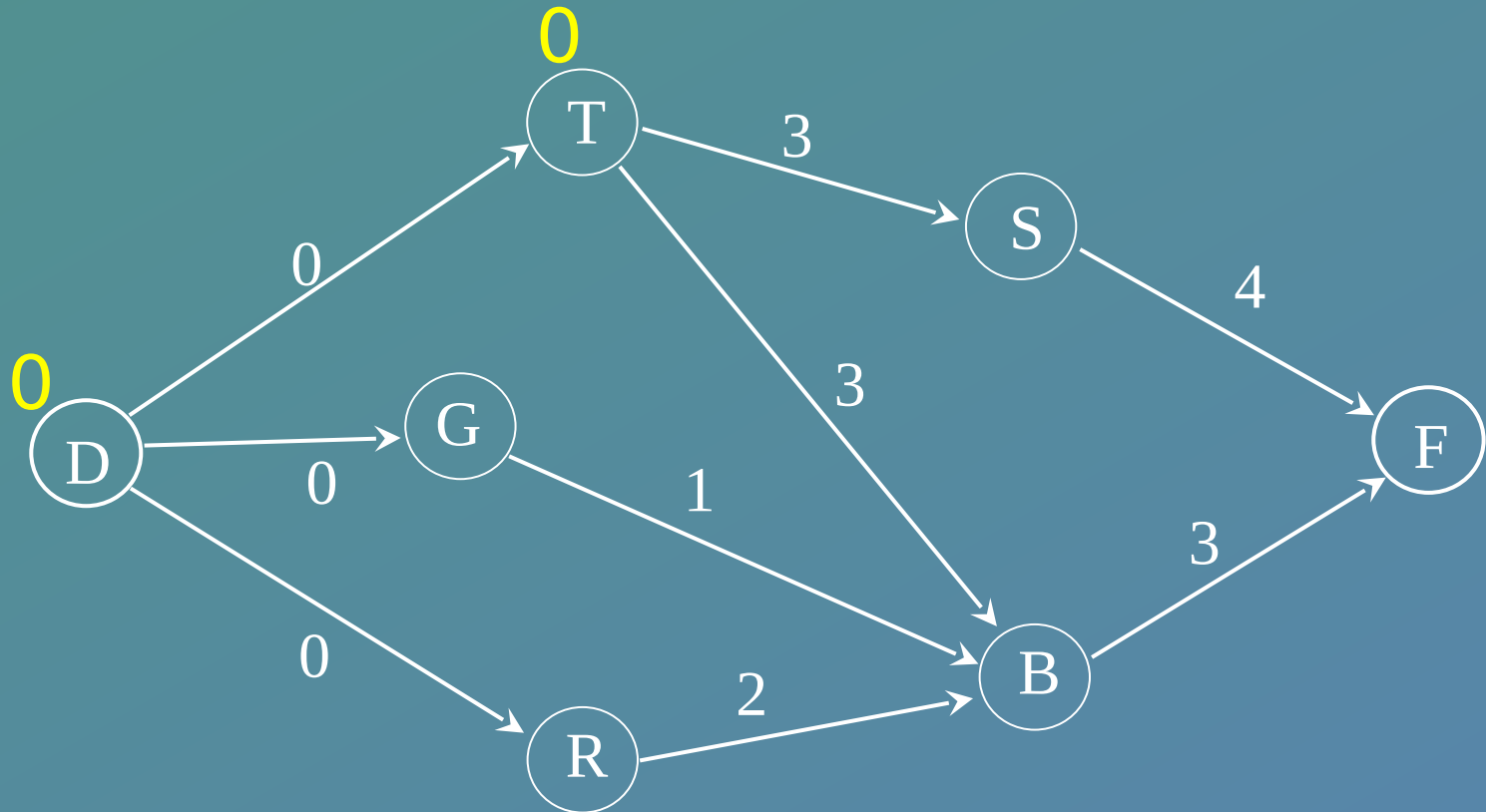
potentiels-tâches



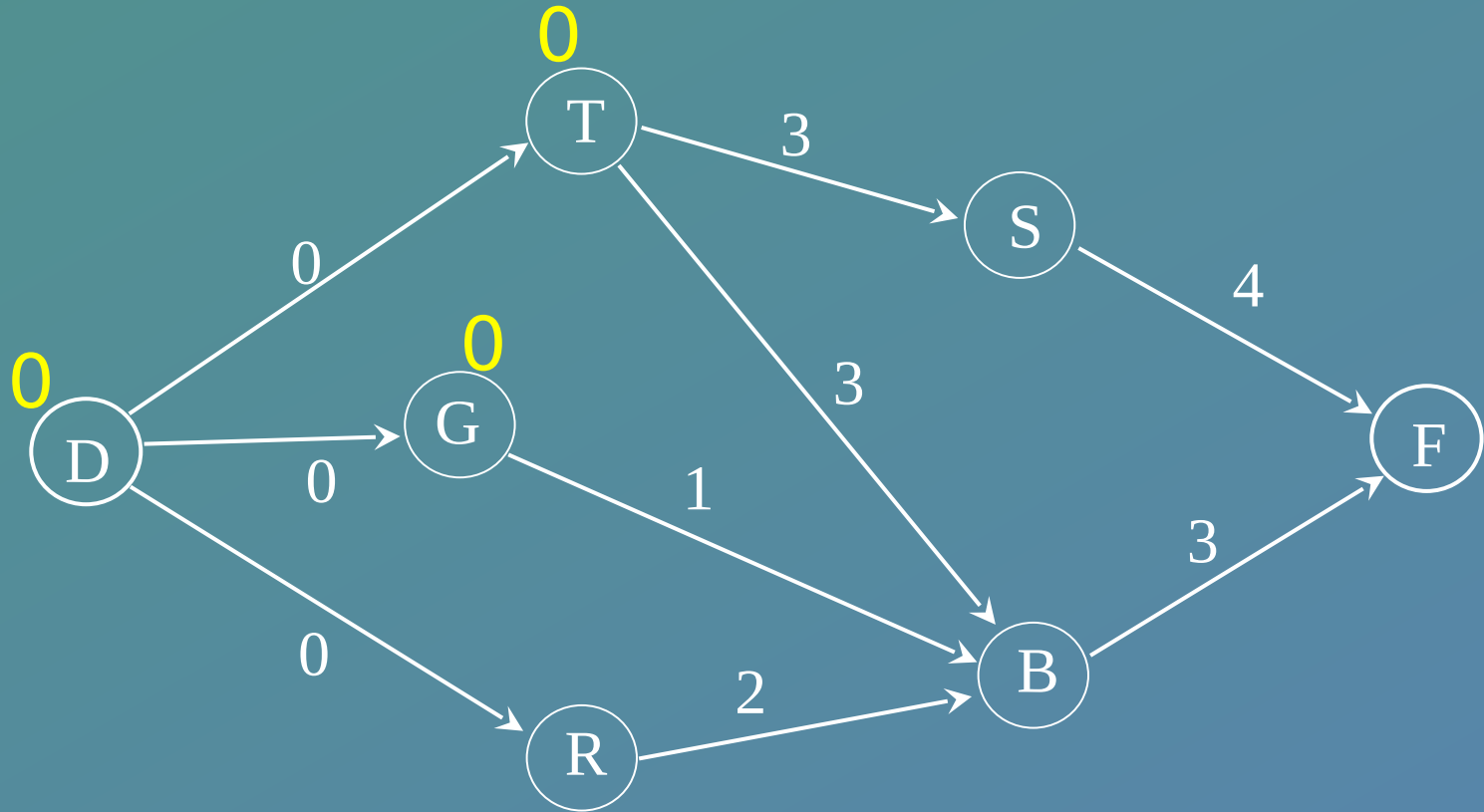
potentiels-tâches



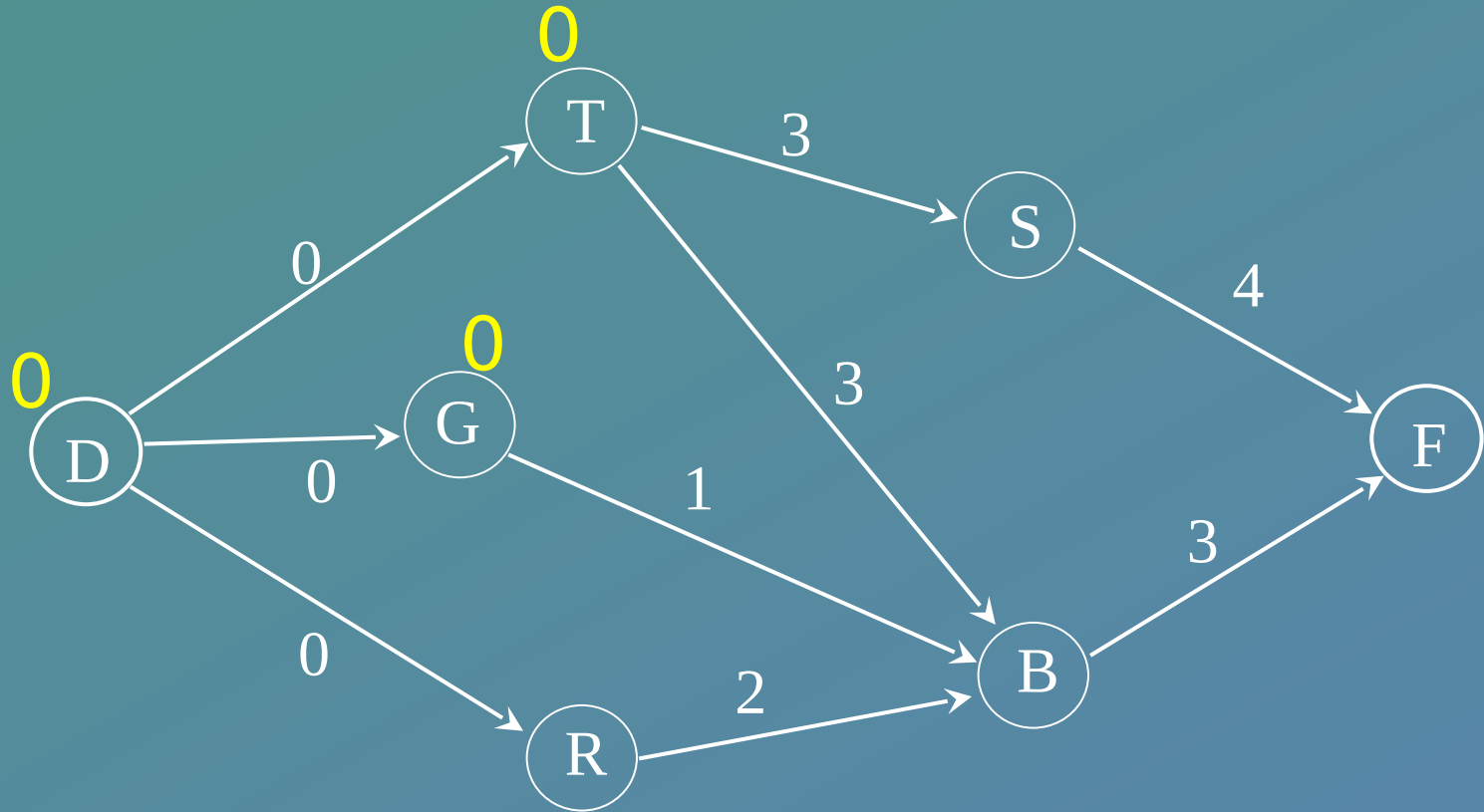
potentiels-tâches



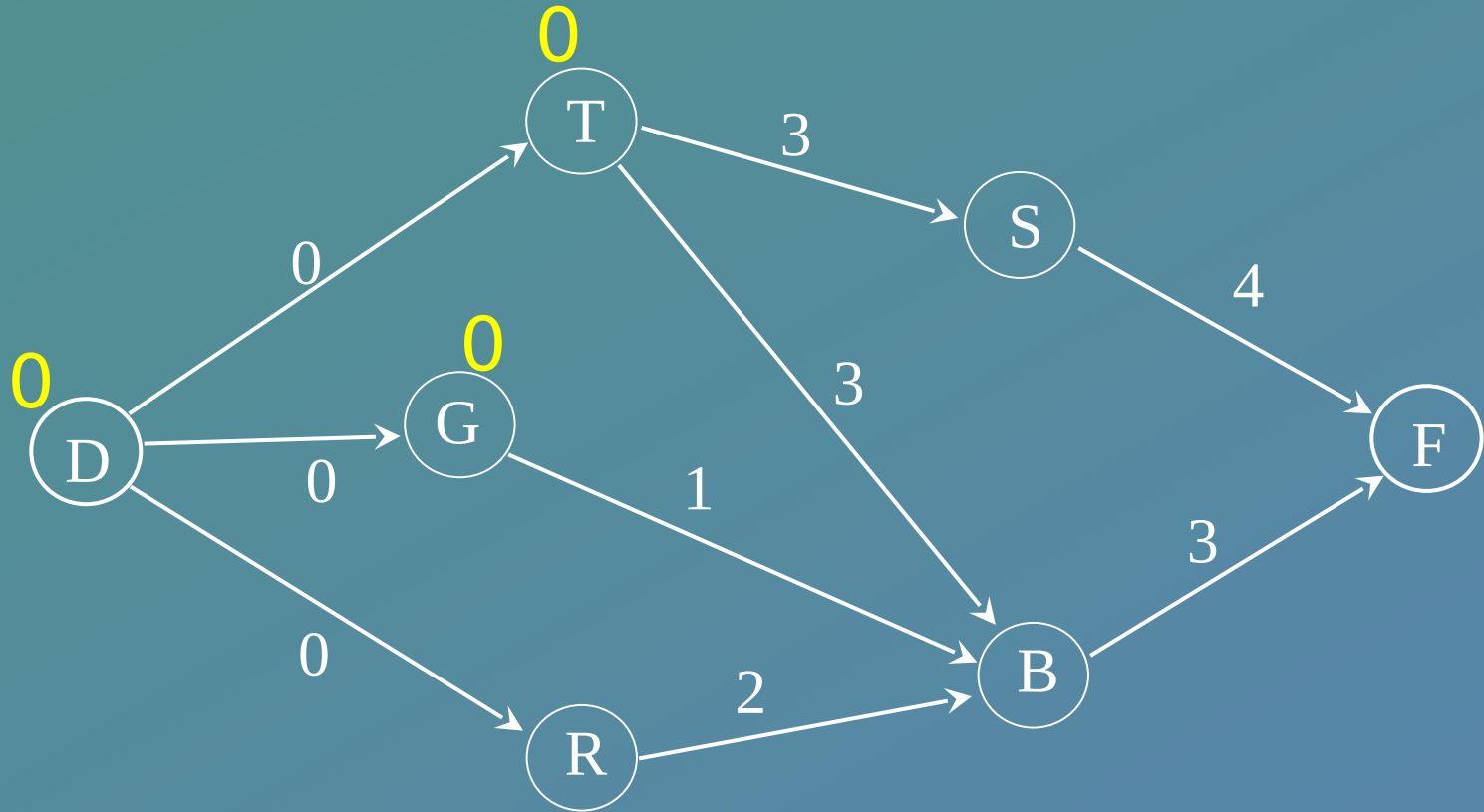
potentiels-tâches



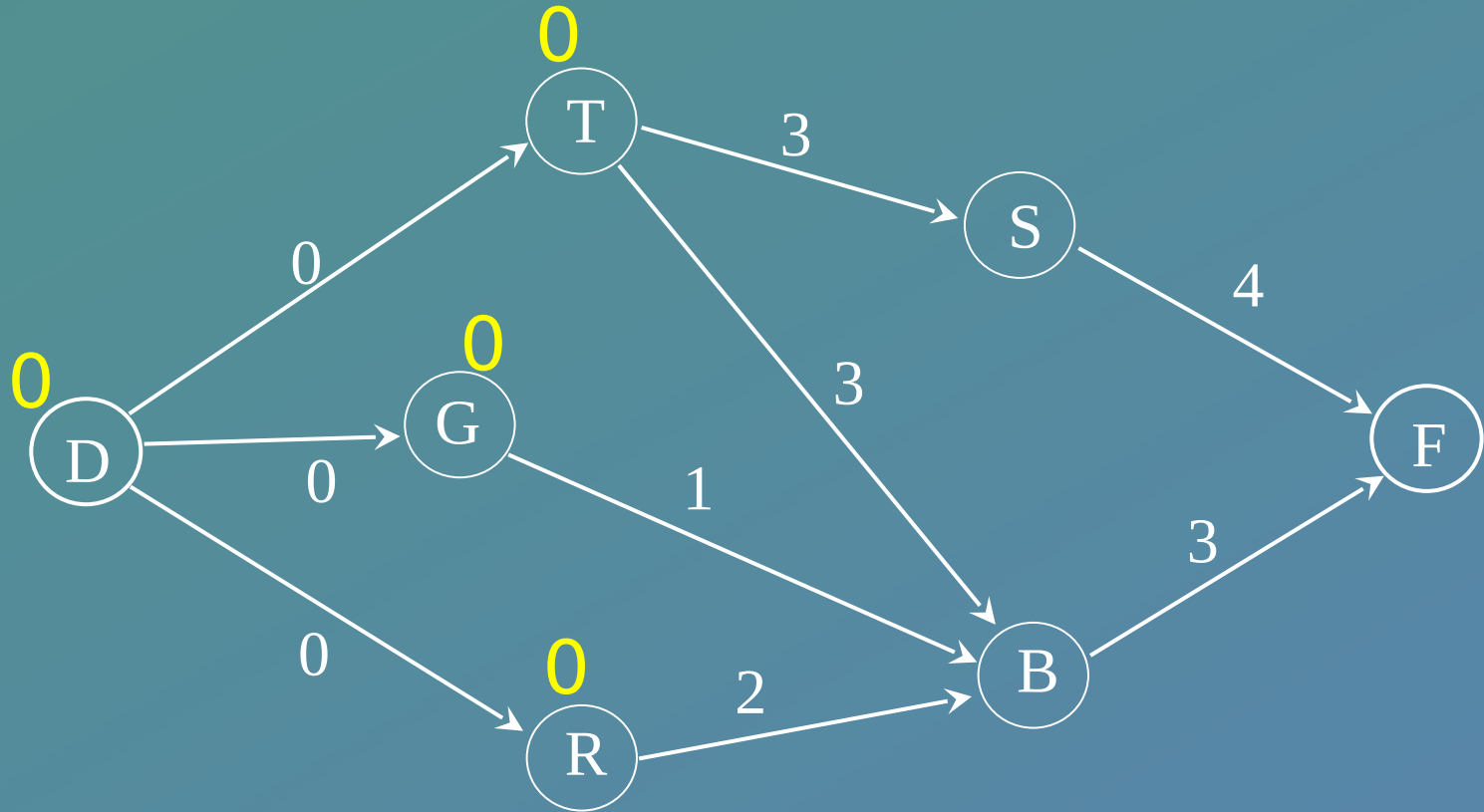
potentiels-tâches



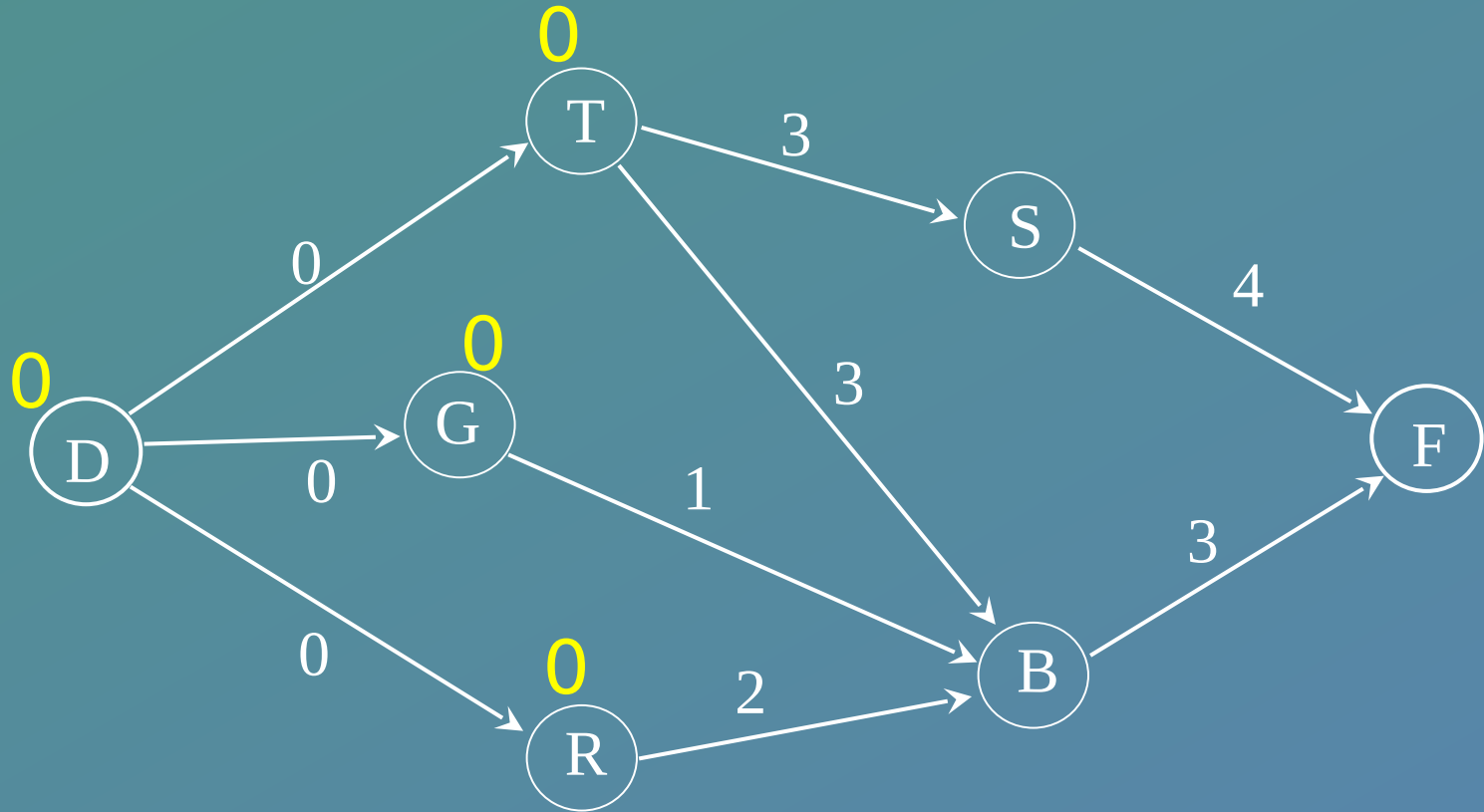
potentiels-tâches



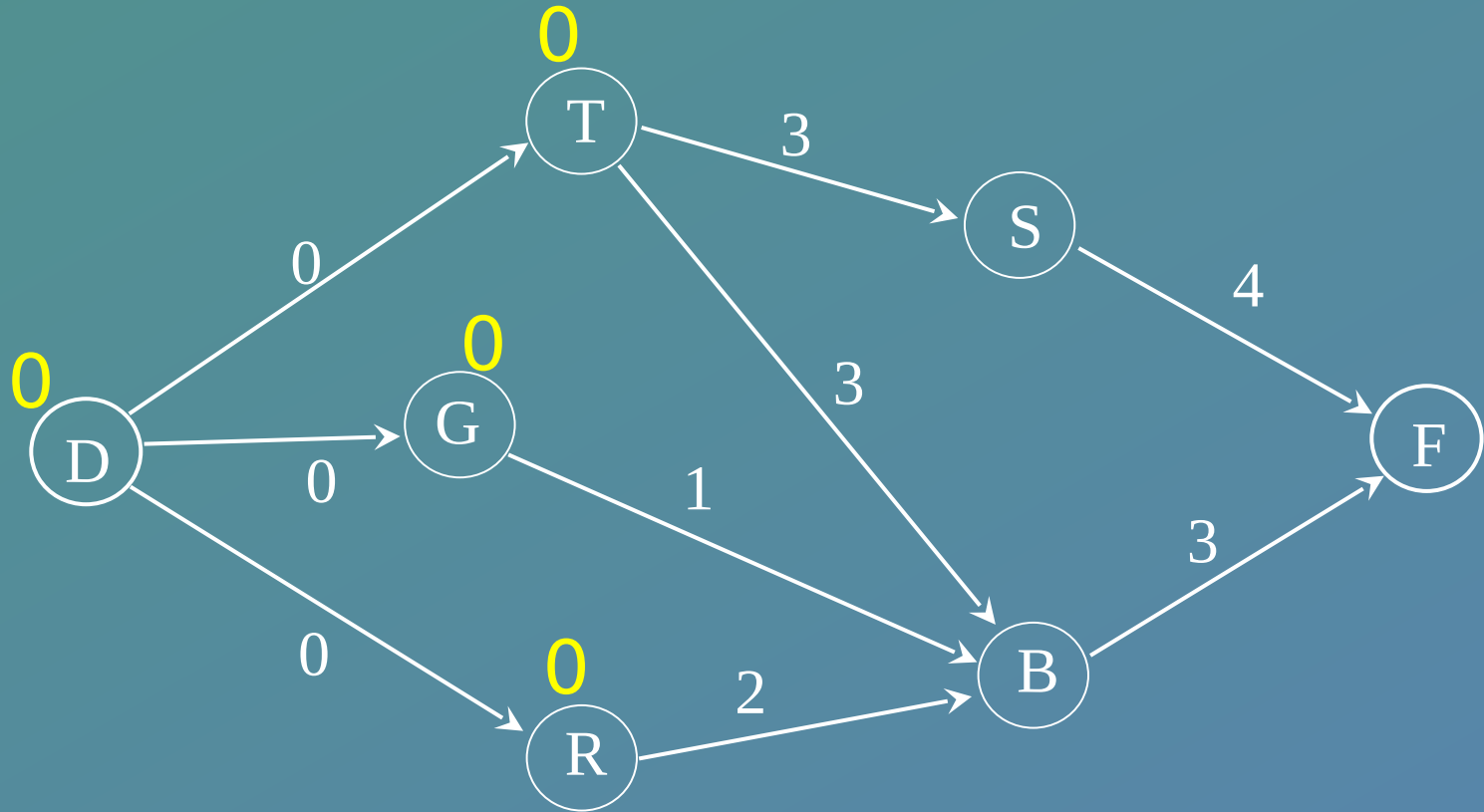
potentiels-tâches



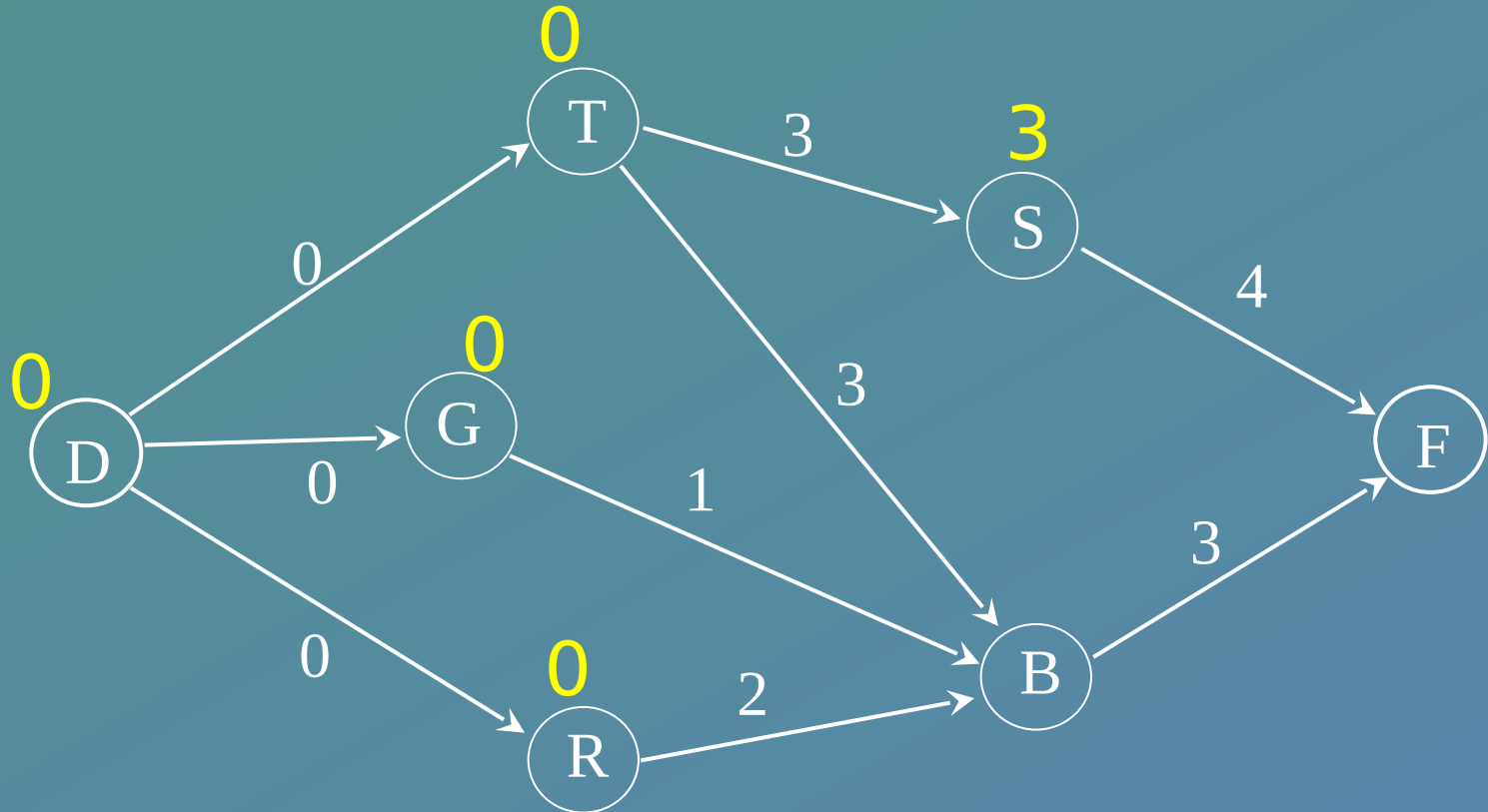
potentiels-tâches



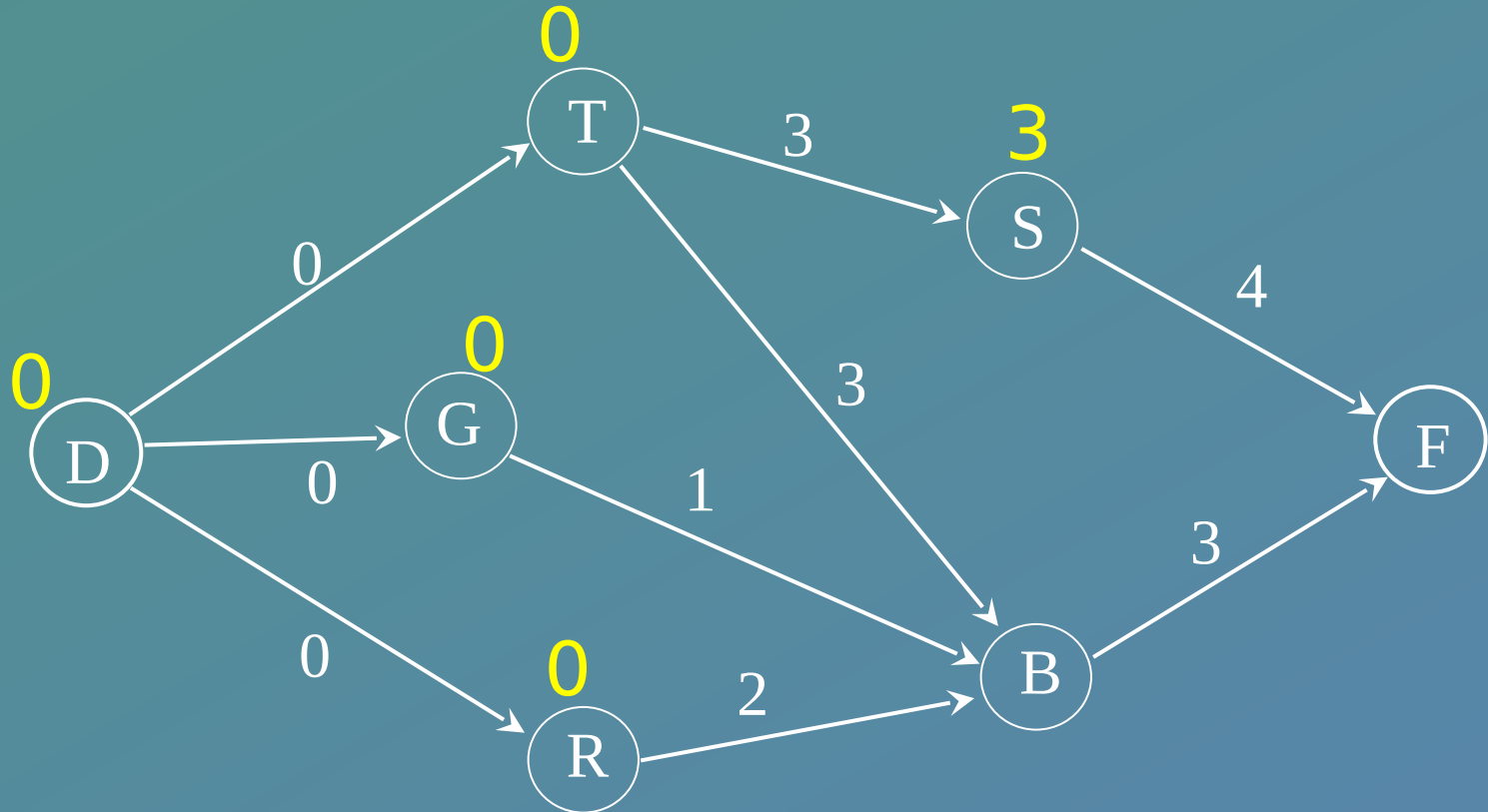
potentiels-tâches



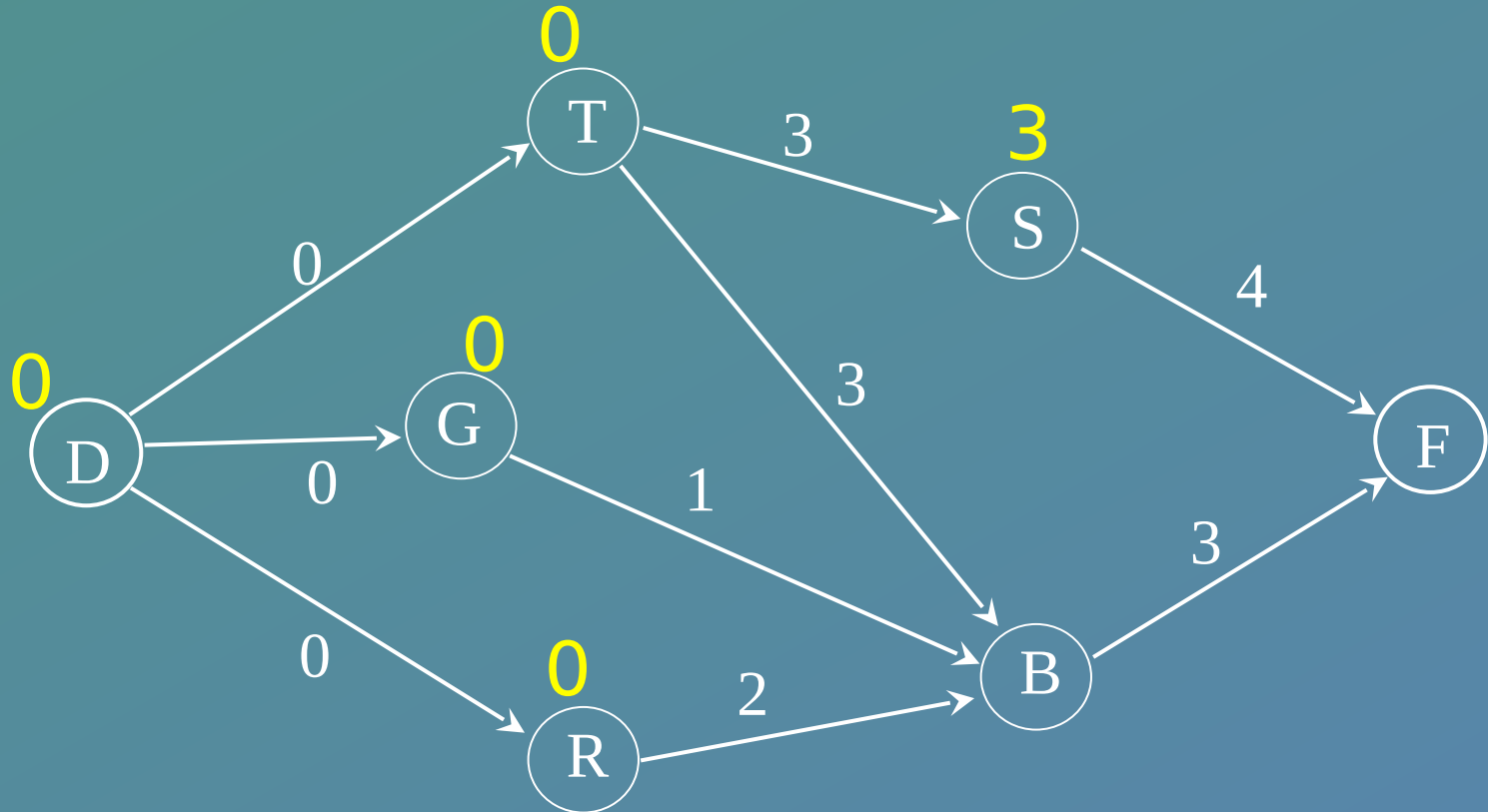
potentiels-tâches



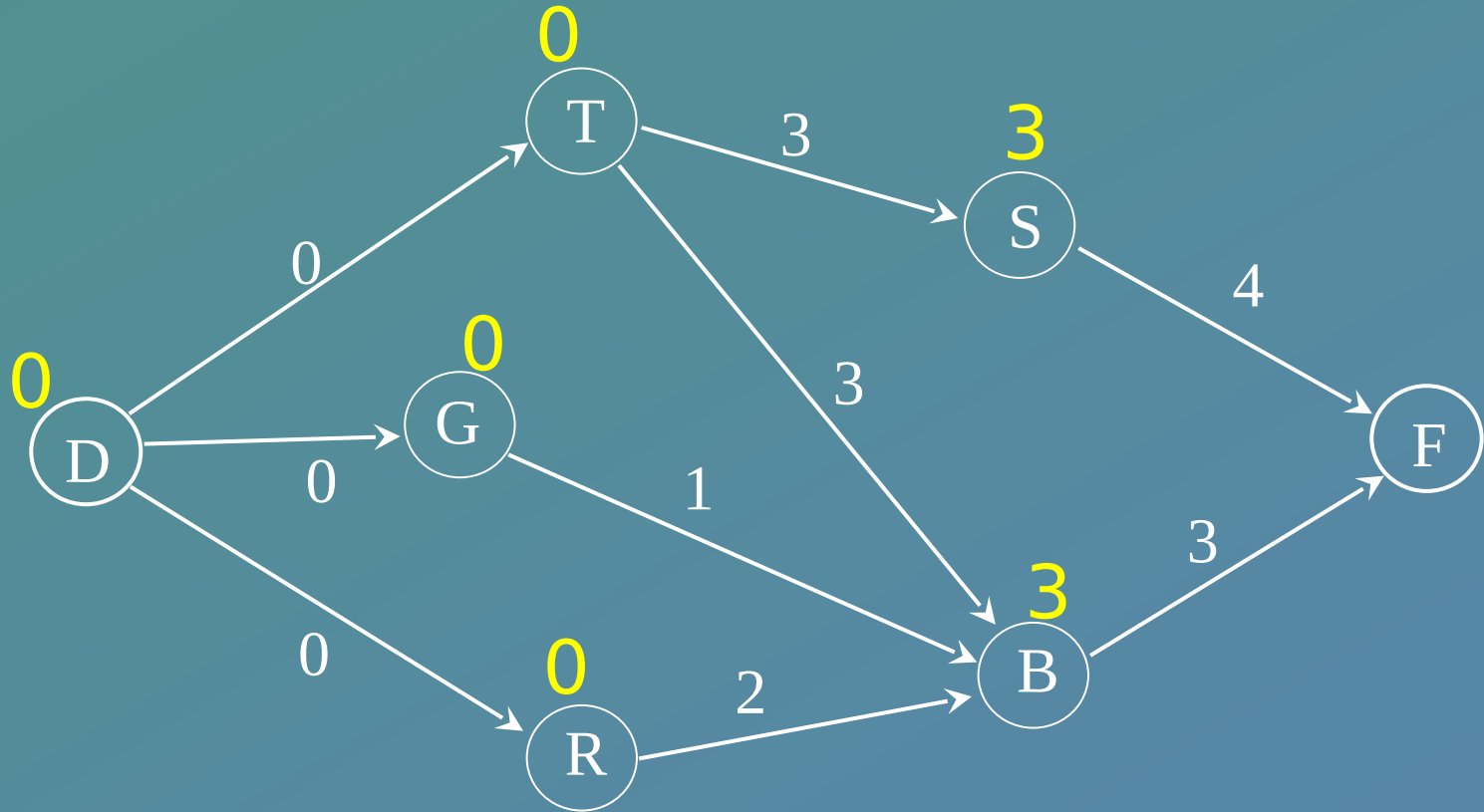
potentiels-tâches



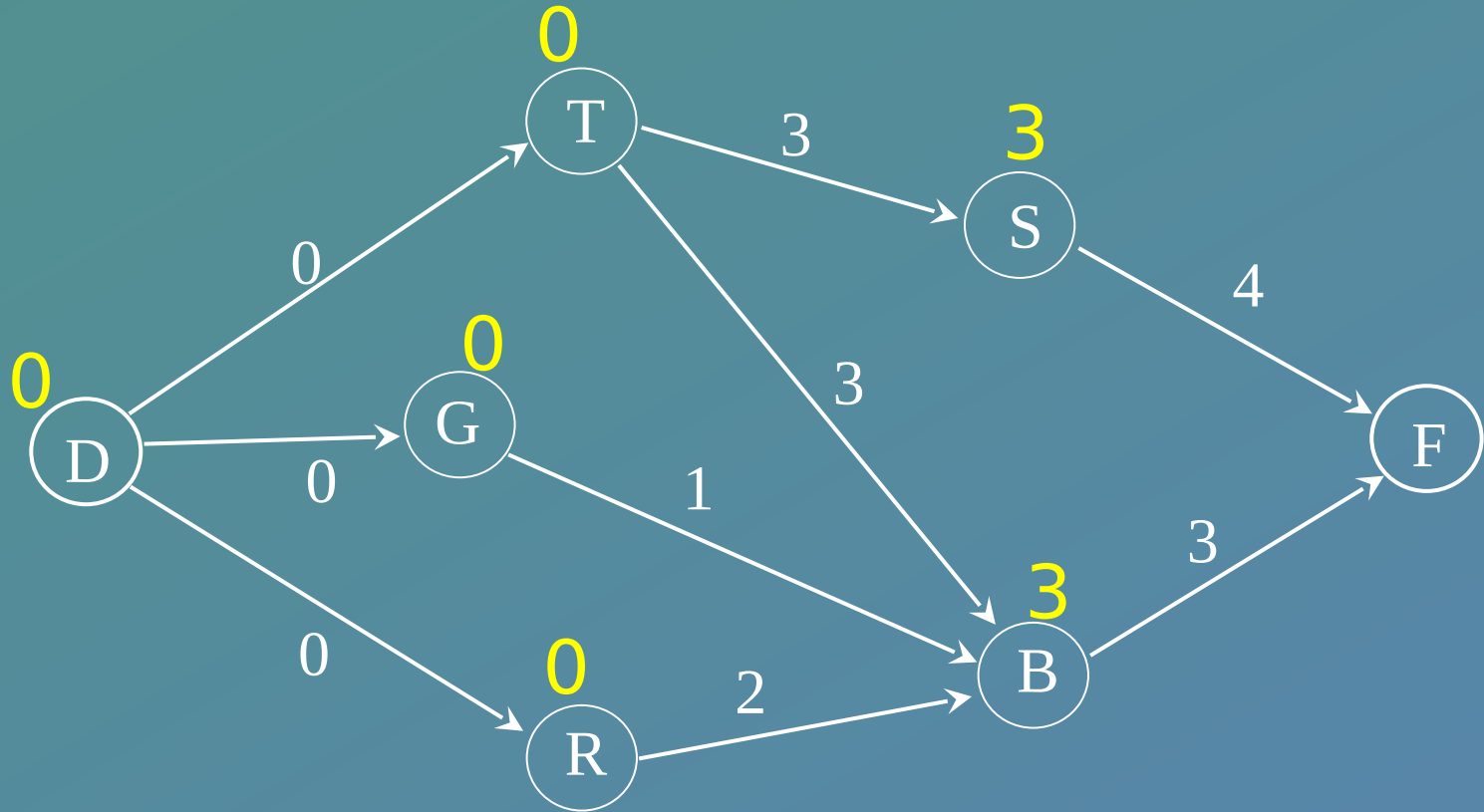
potentiels-tâches



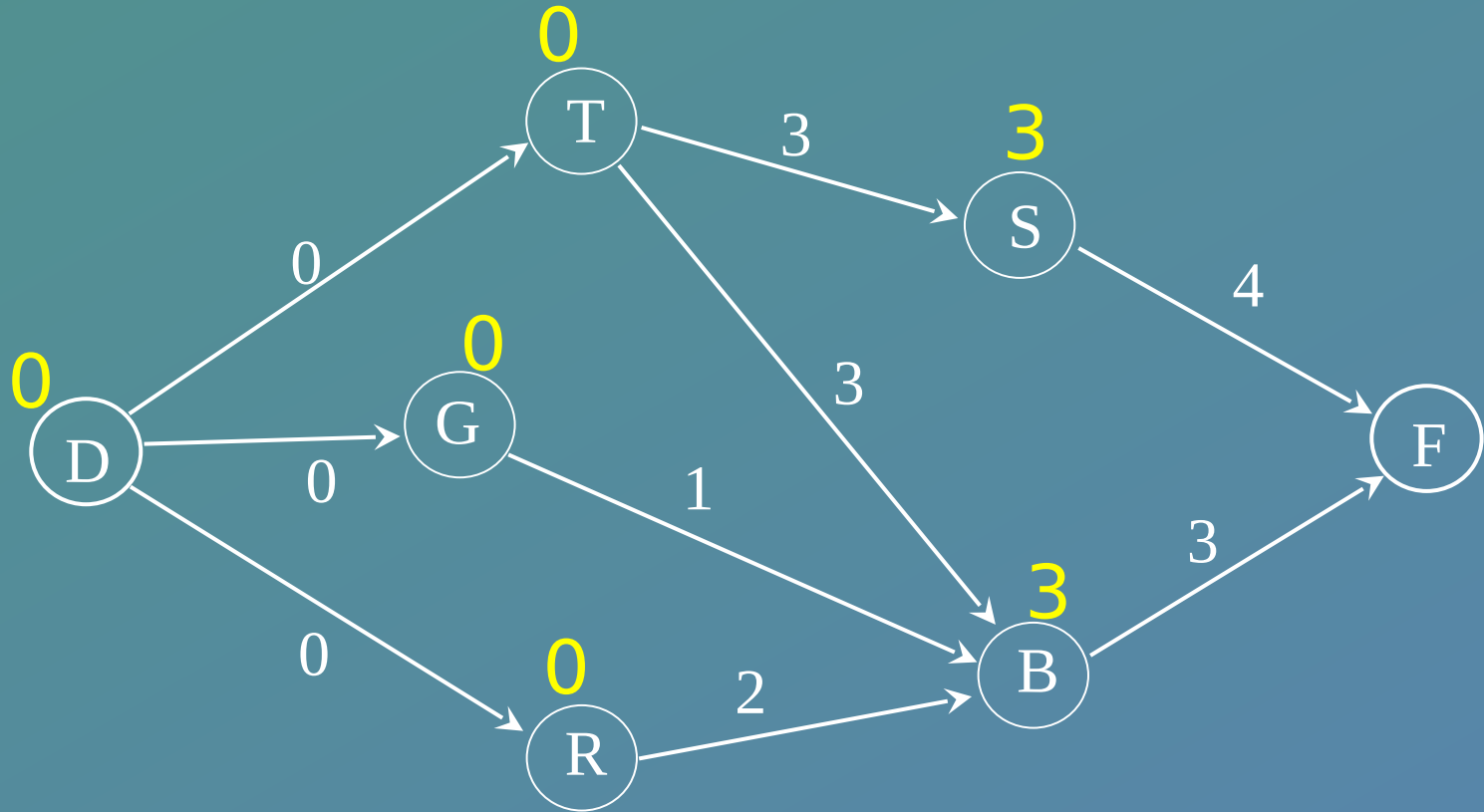
potentiels-tâches



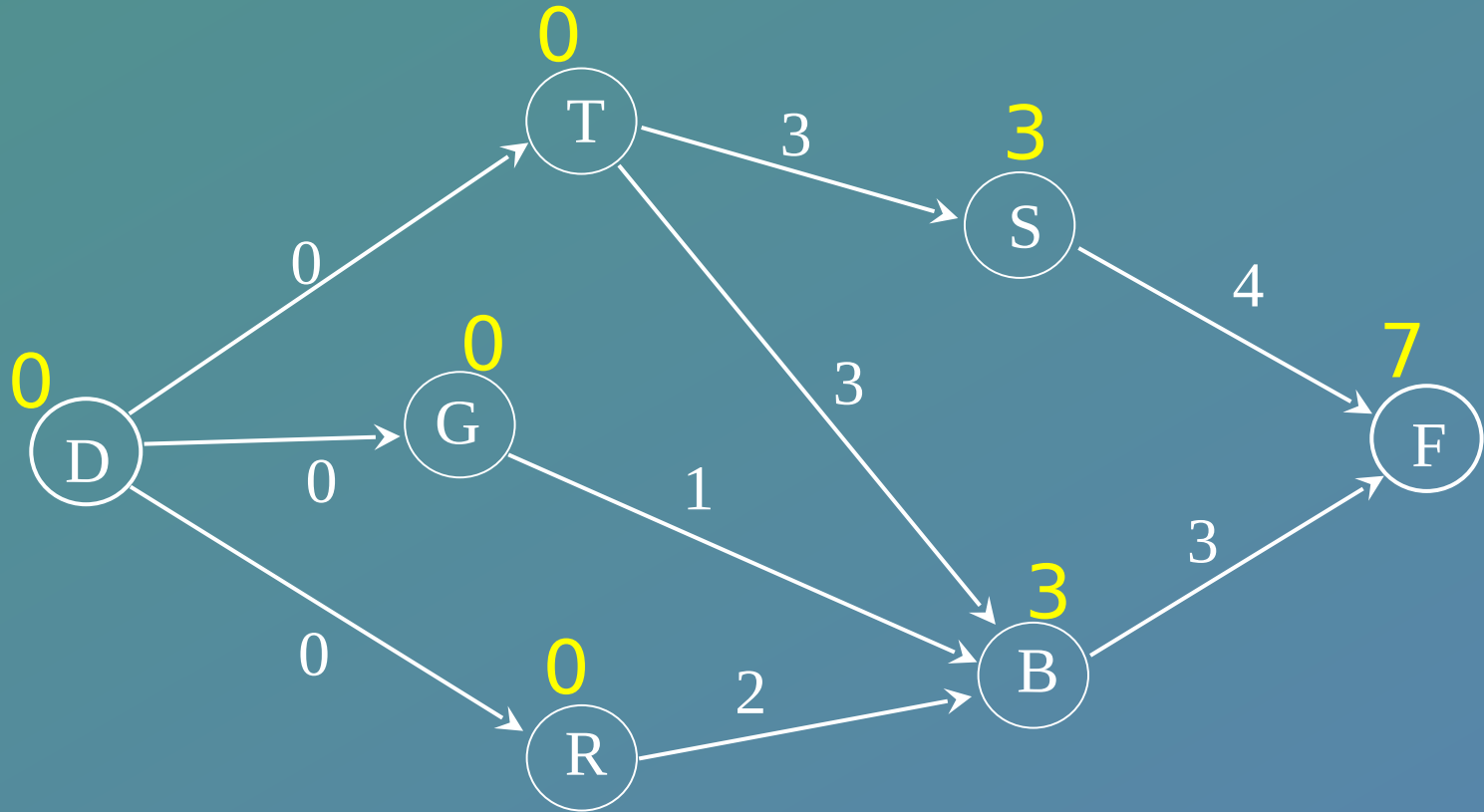
potentiels-tâches



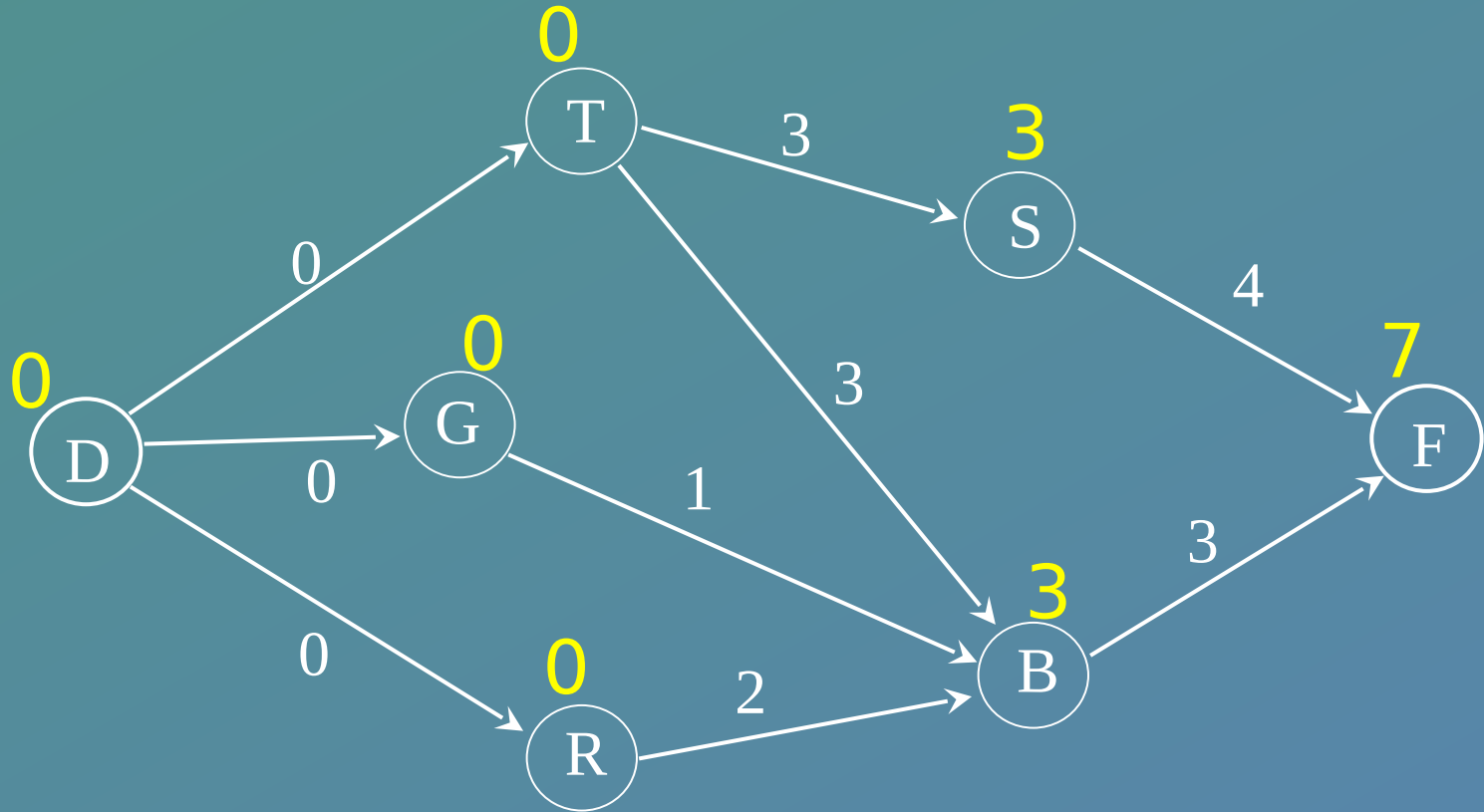
potentiels-tâches



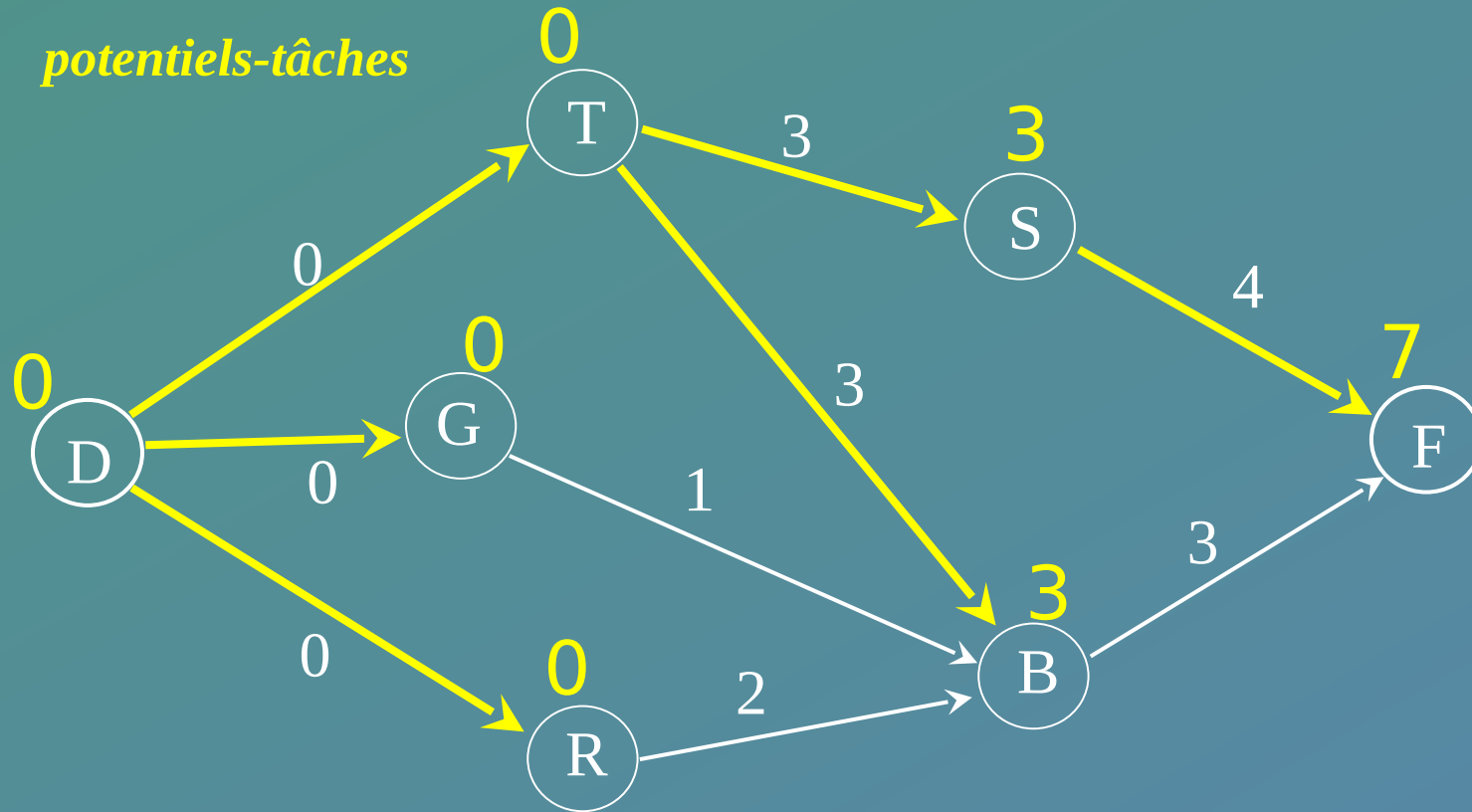
potentiels-tâches



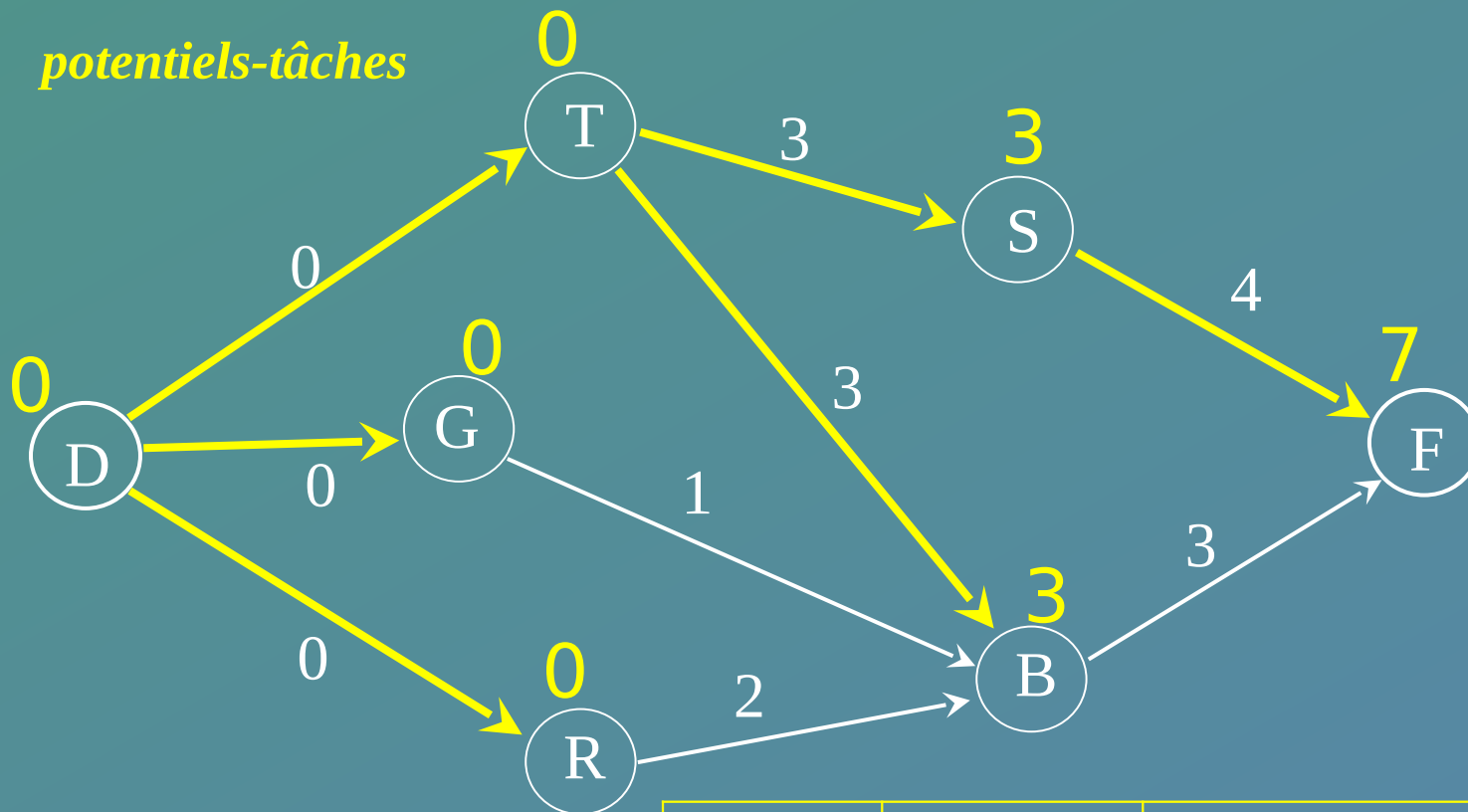
potentiels-tâches



potentiels-tâches



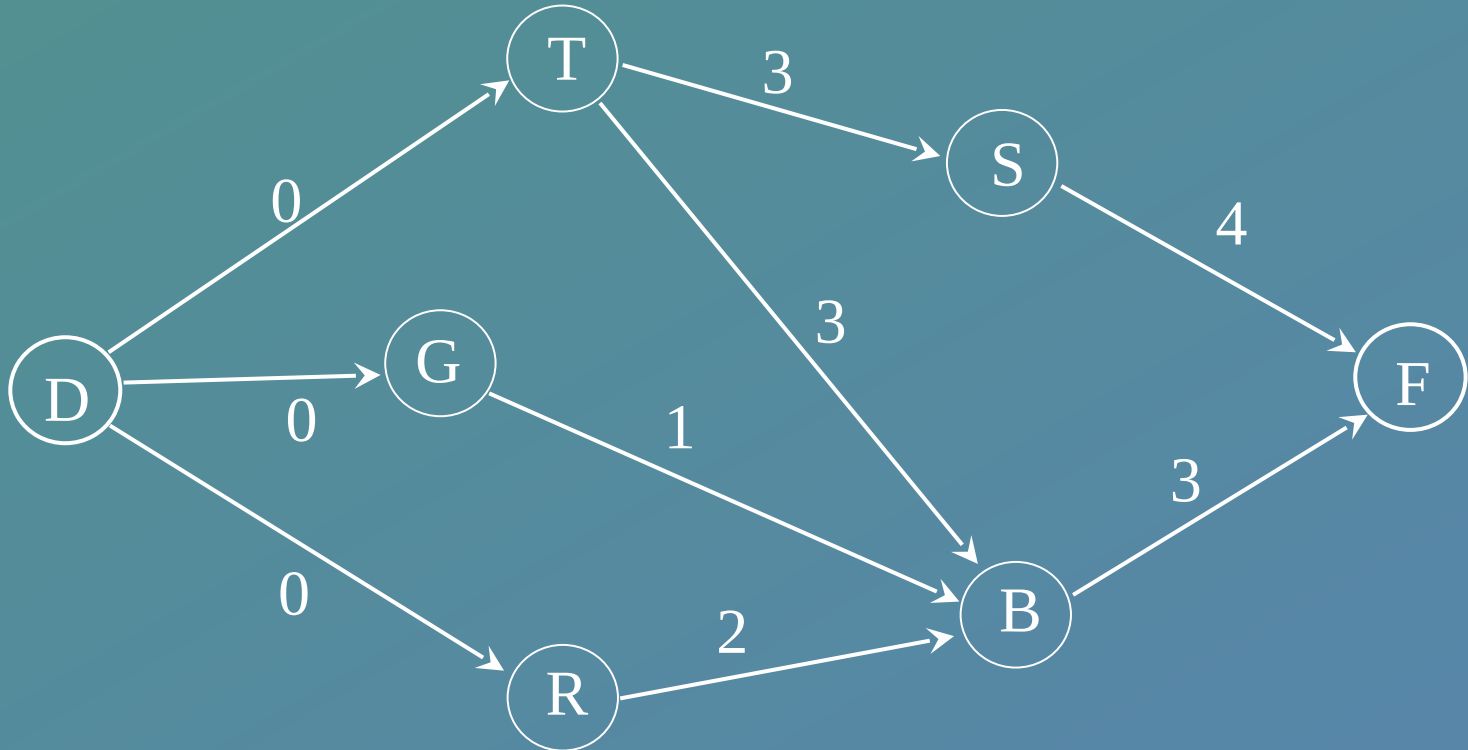
potentiels-tâches



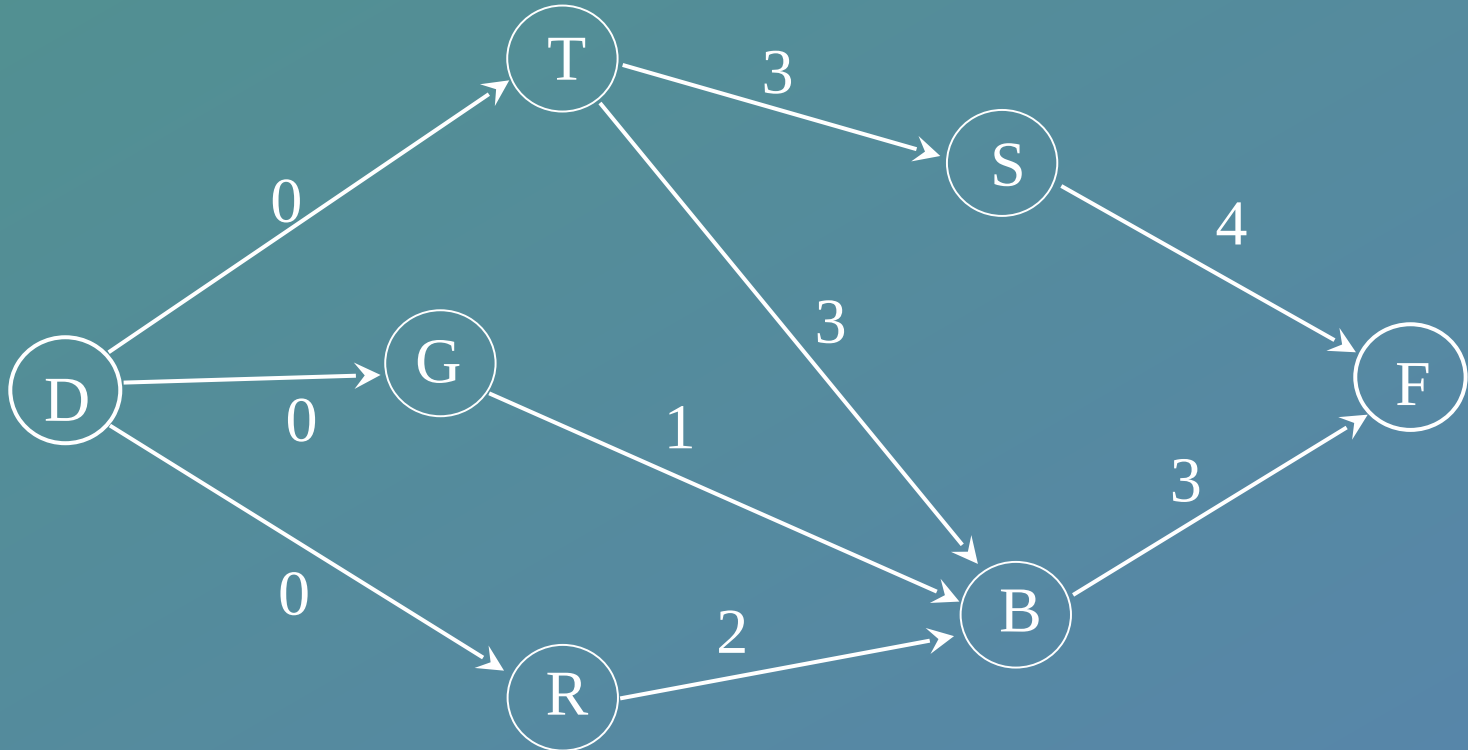
tâche	durée	après :	la date au plus tôt
Début	0	—	0
T	3	Début	0
G	1	Début	0
R	2	Début	0
B	3	T, G, R	3
S	4	T	3
Fin	0	T, G, R, B, S	7



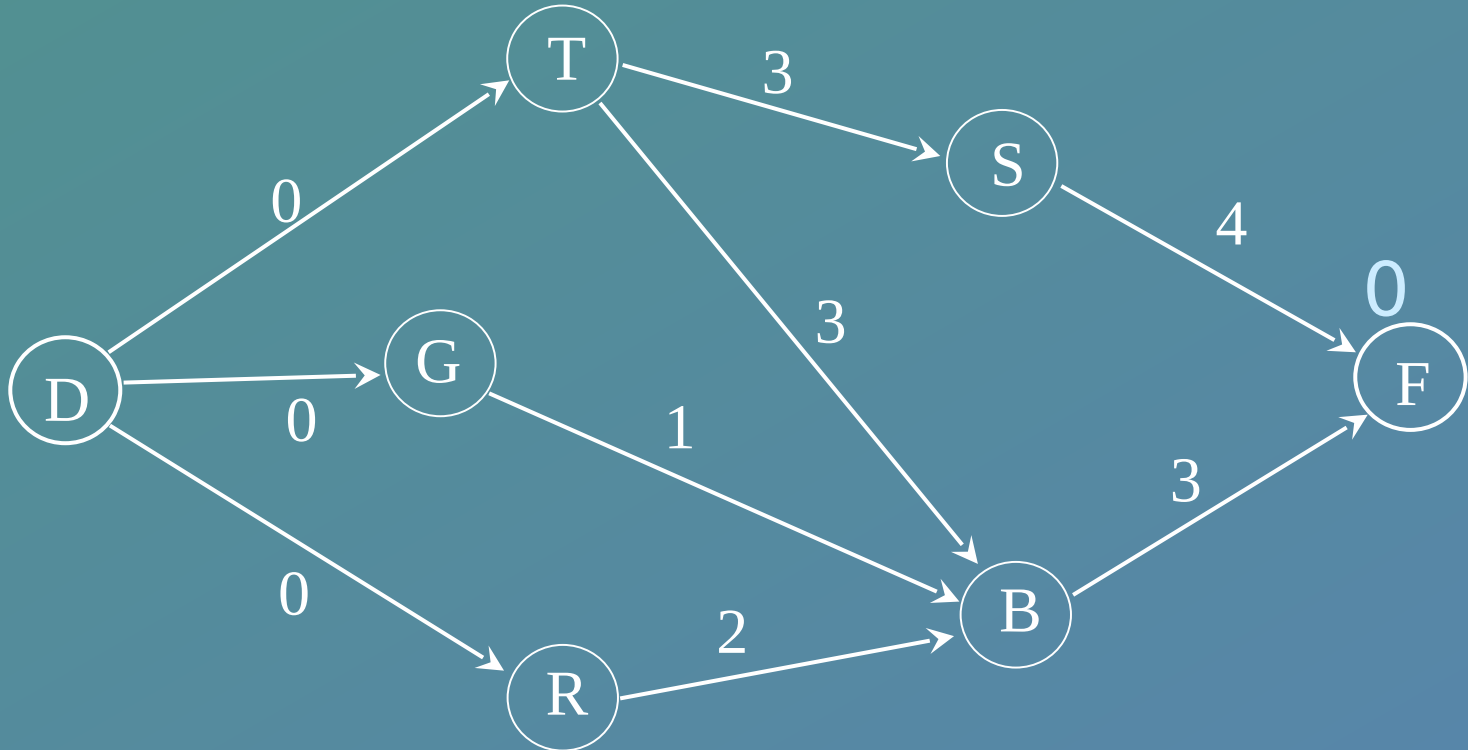
potentiels-tâches



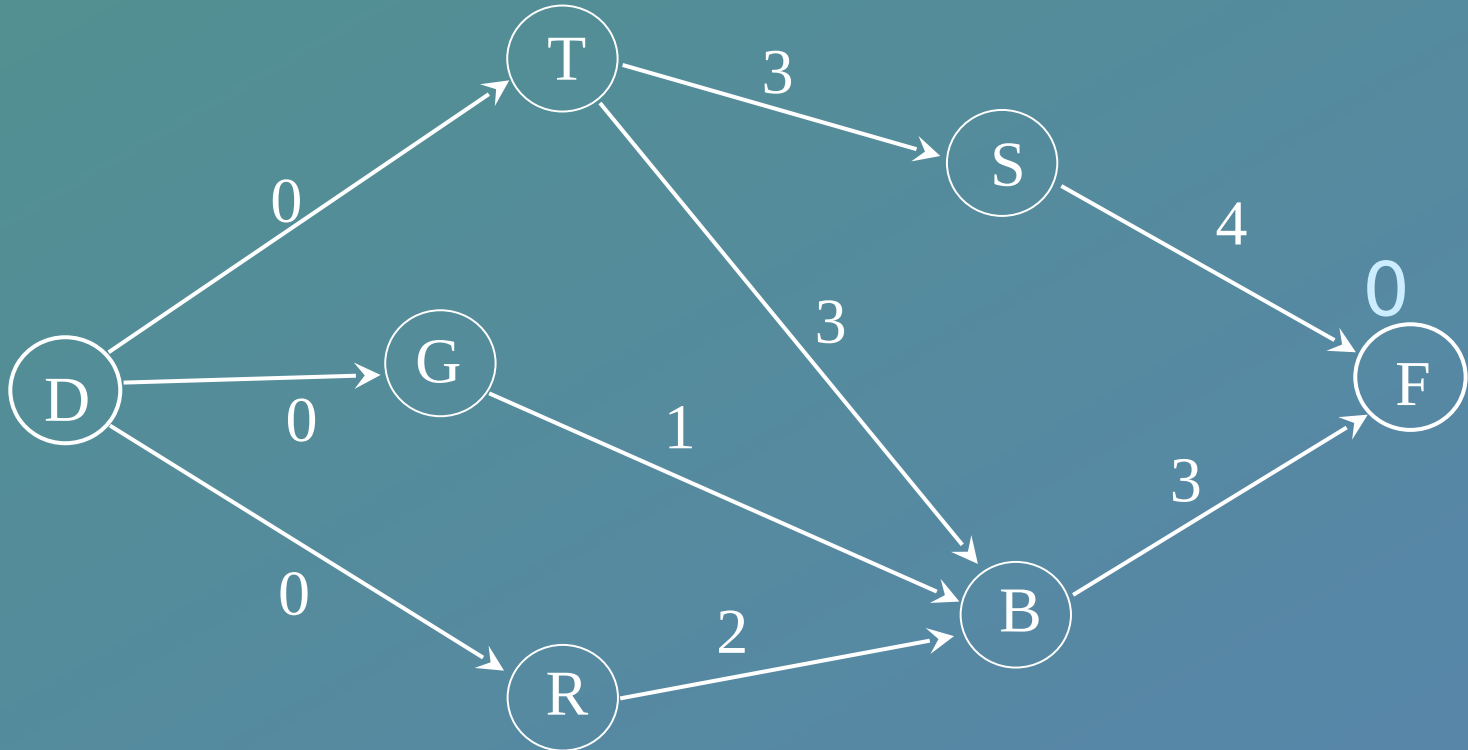
potentiels-tâches



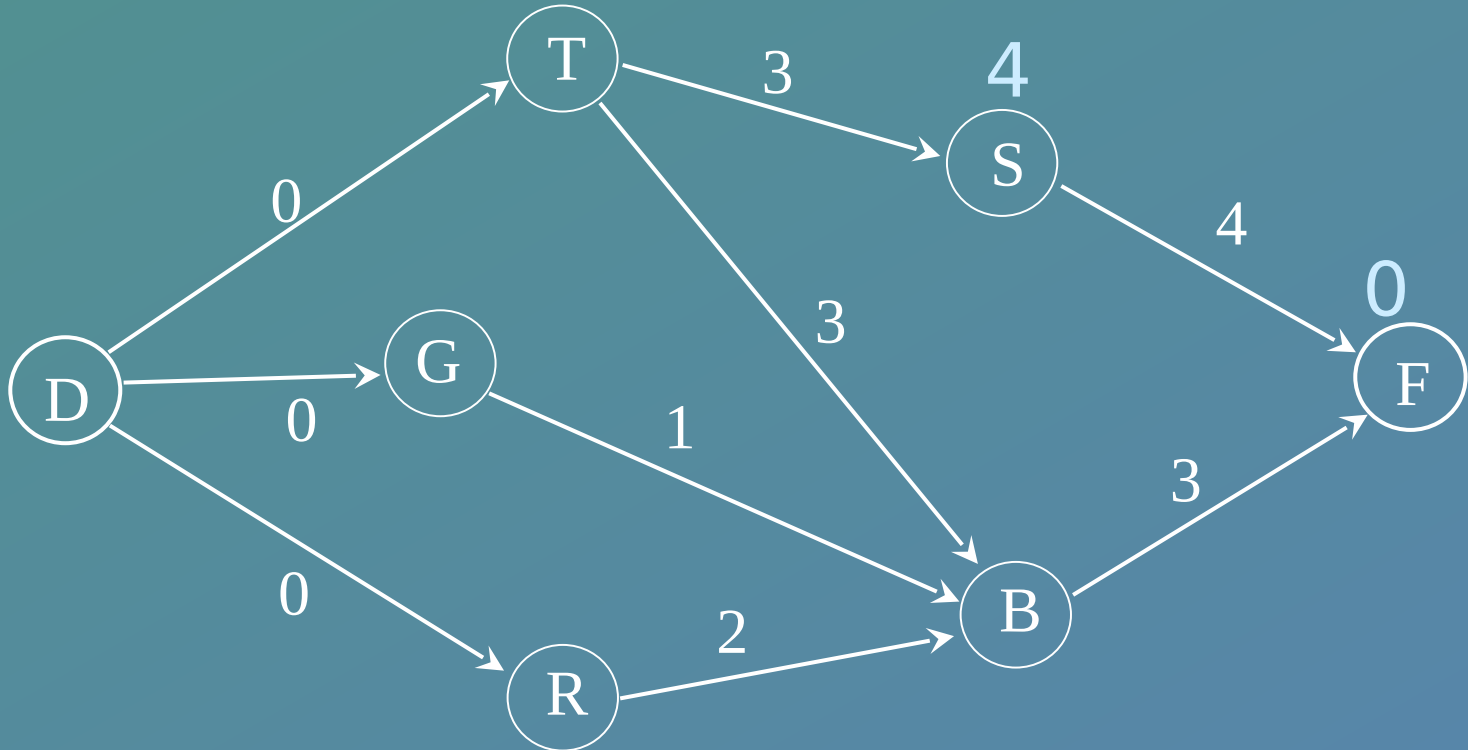
potentiels-tâches



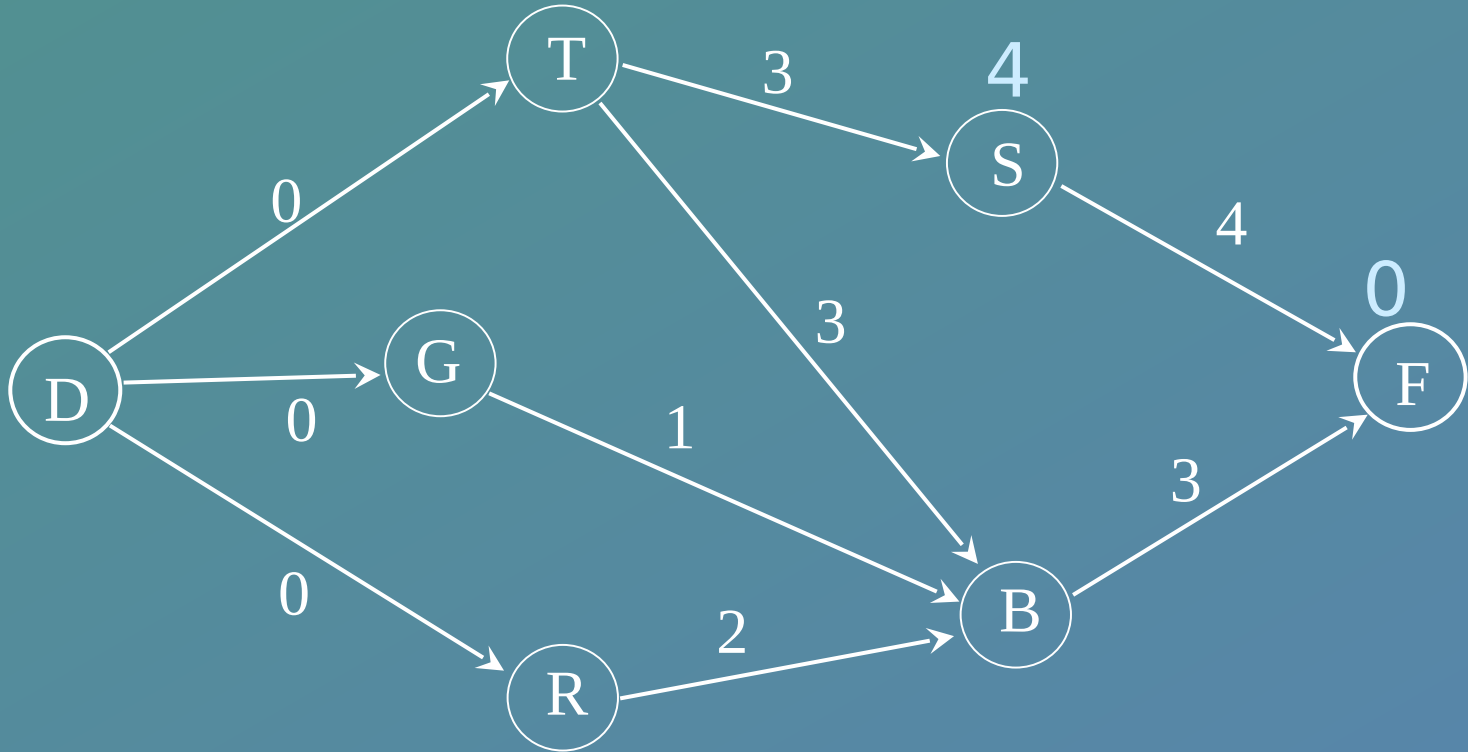
potentiels-tâches



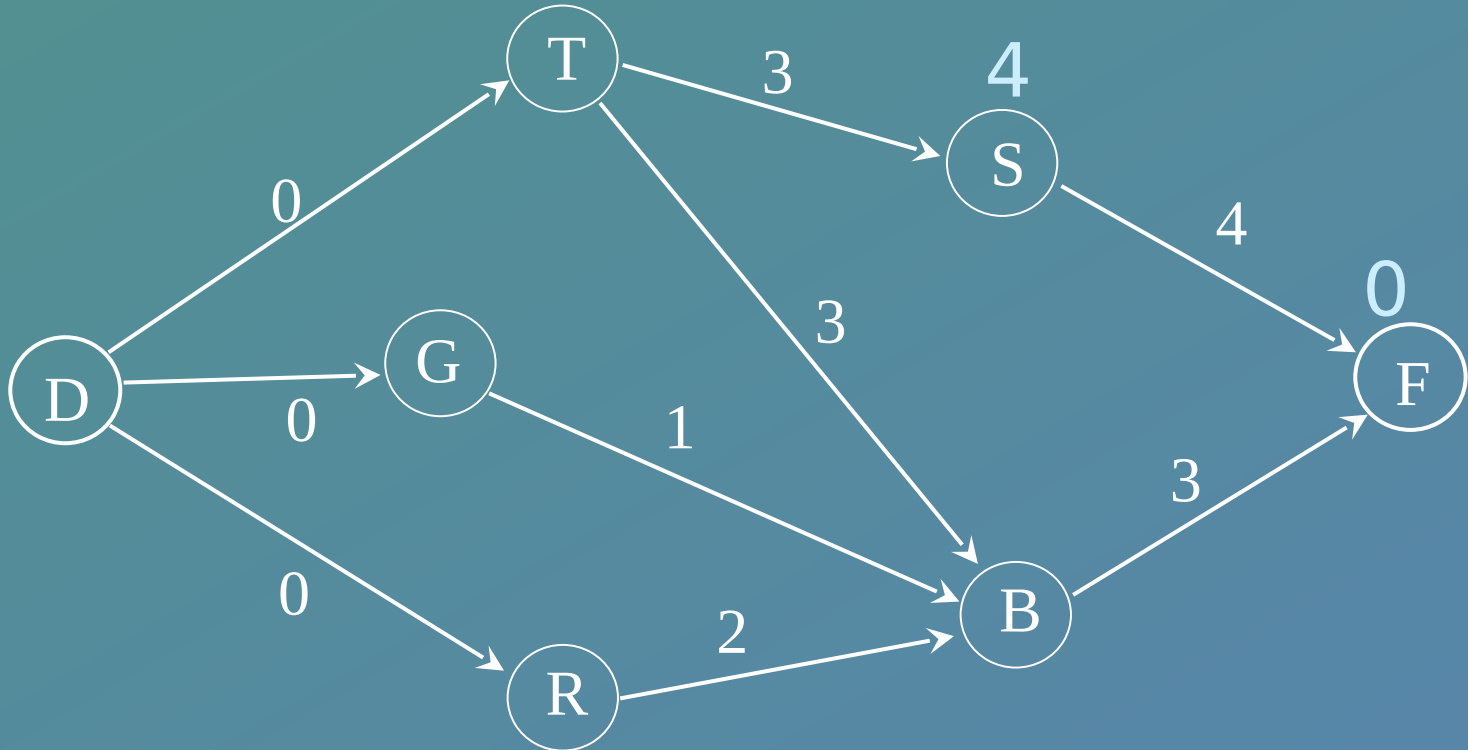
potentiels-tâches



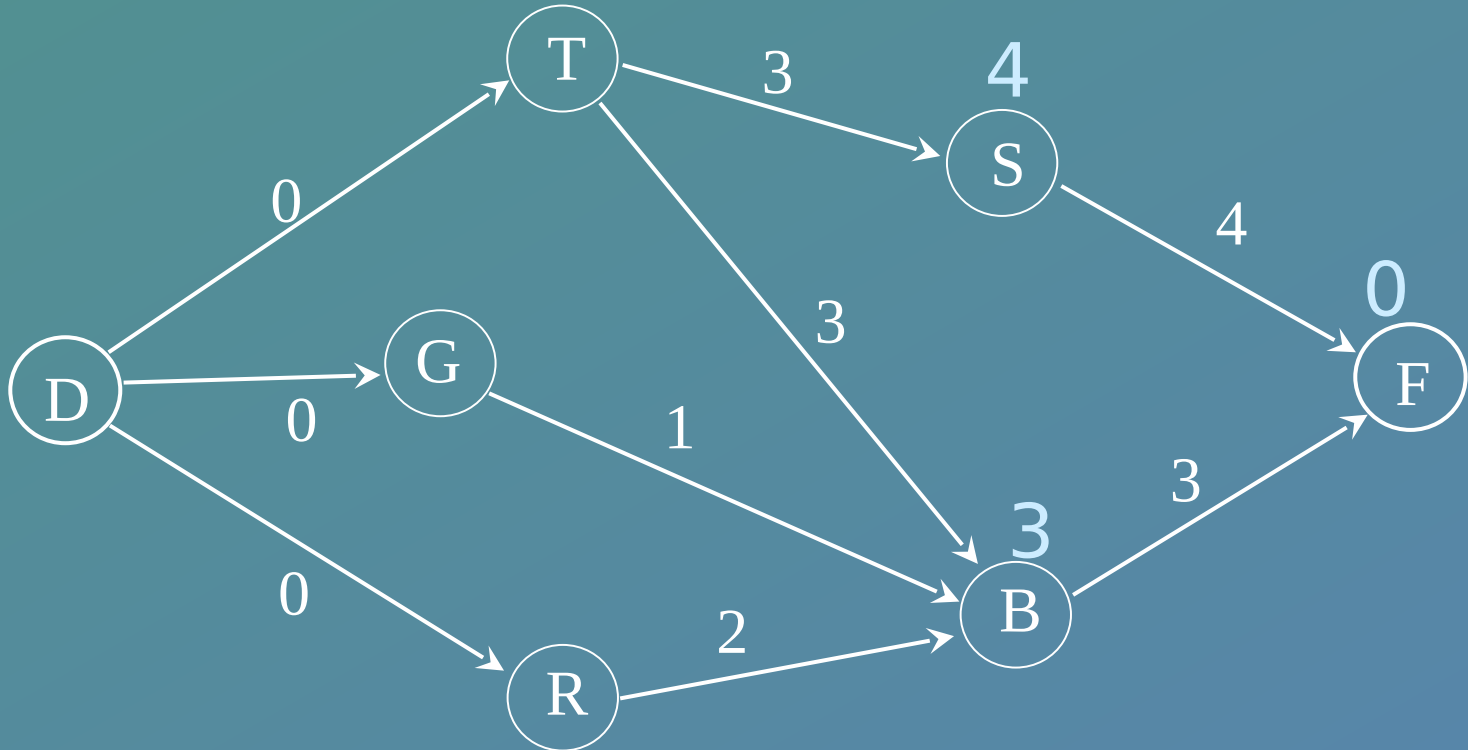
potentiels-tâches



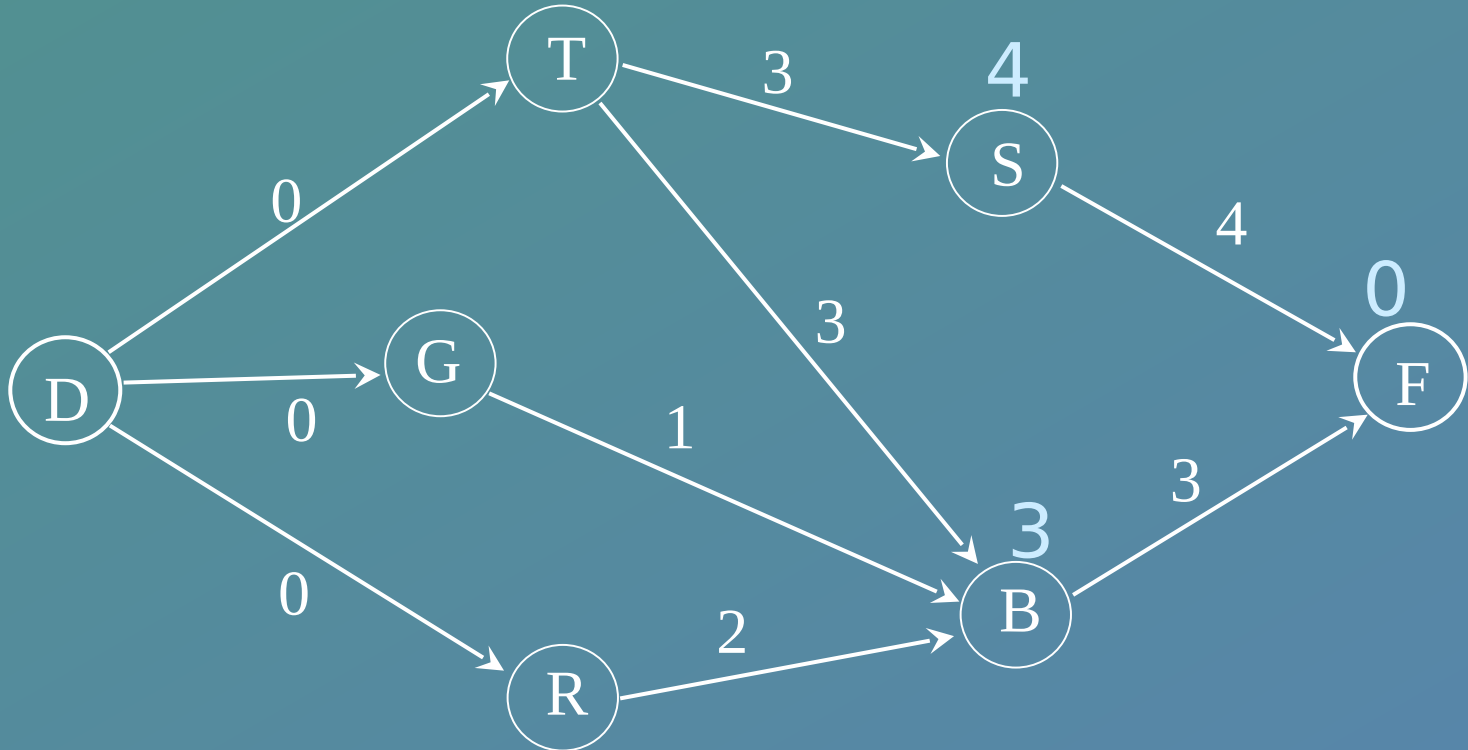
potentiels-tâches



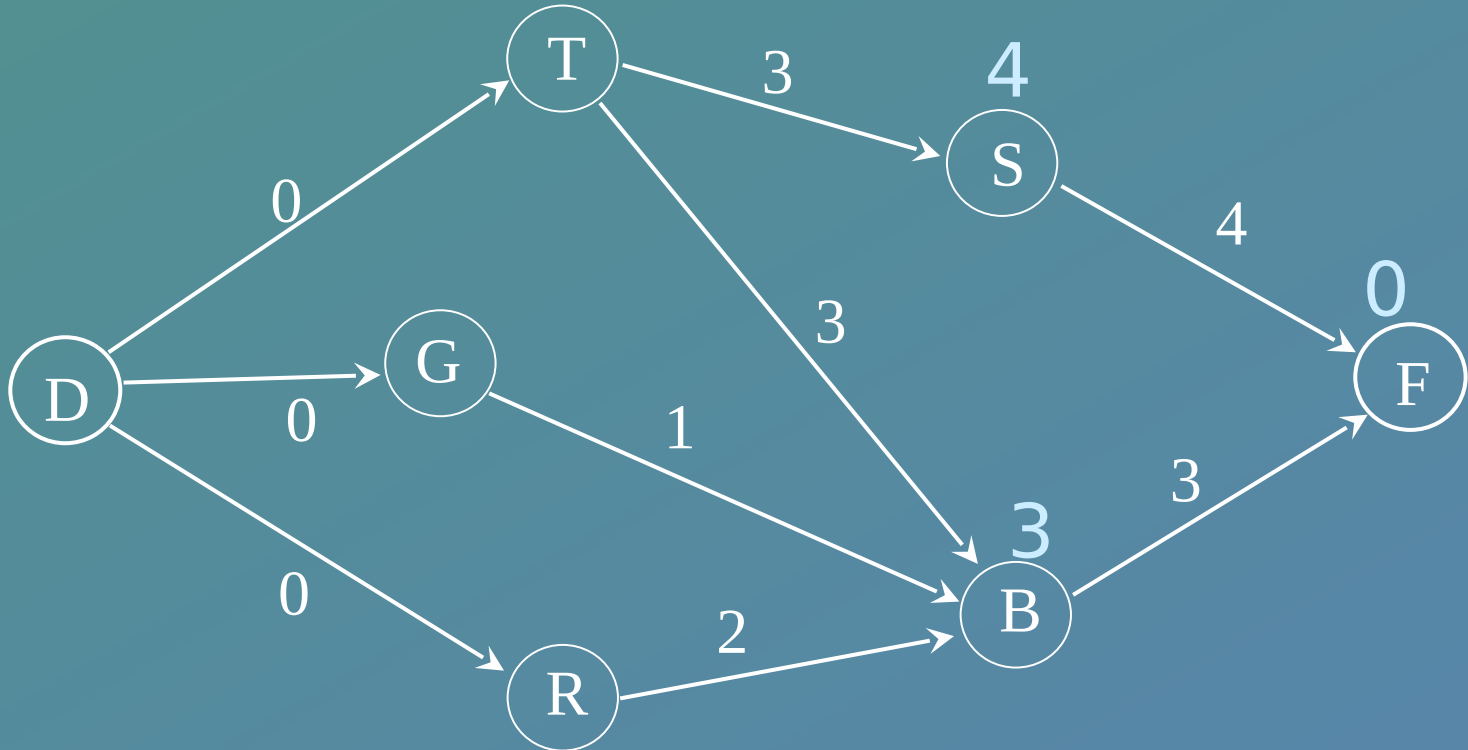
potentiels-tâches



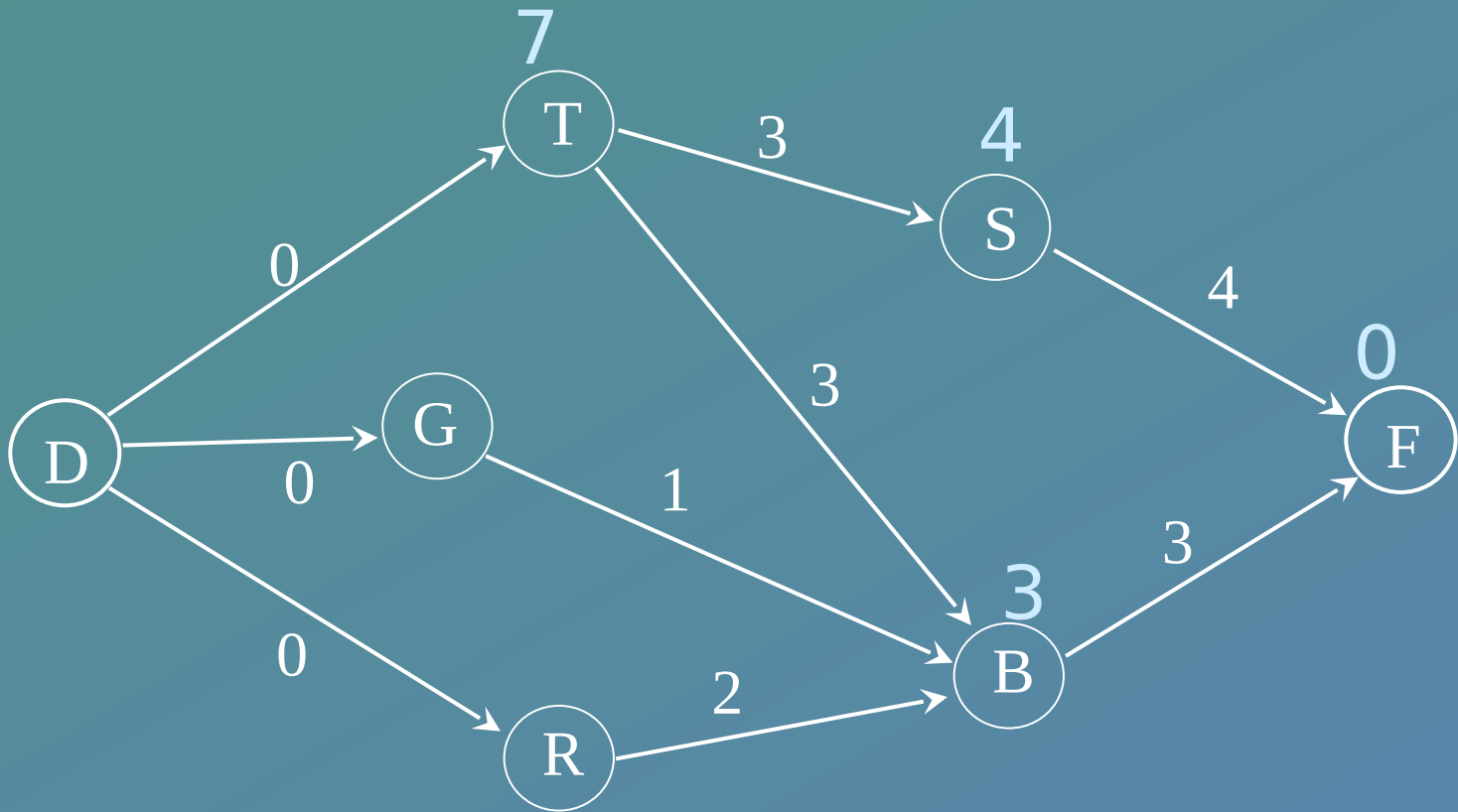
potentiels-tâches



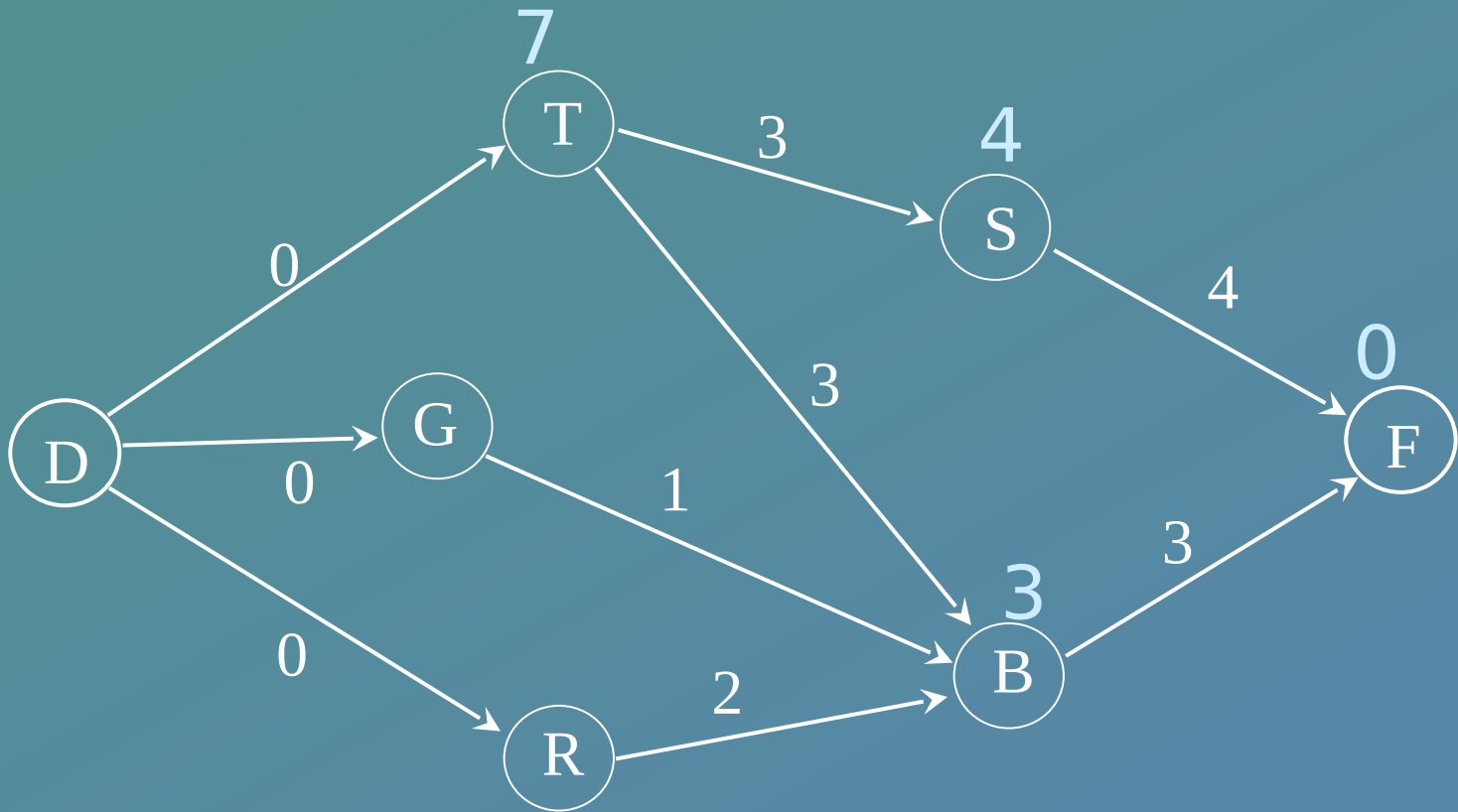
potentiels-tâches



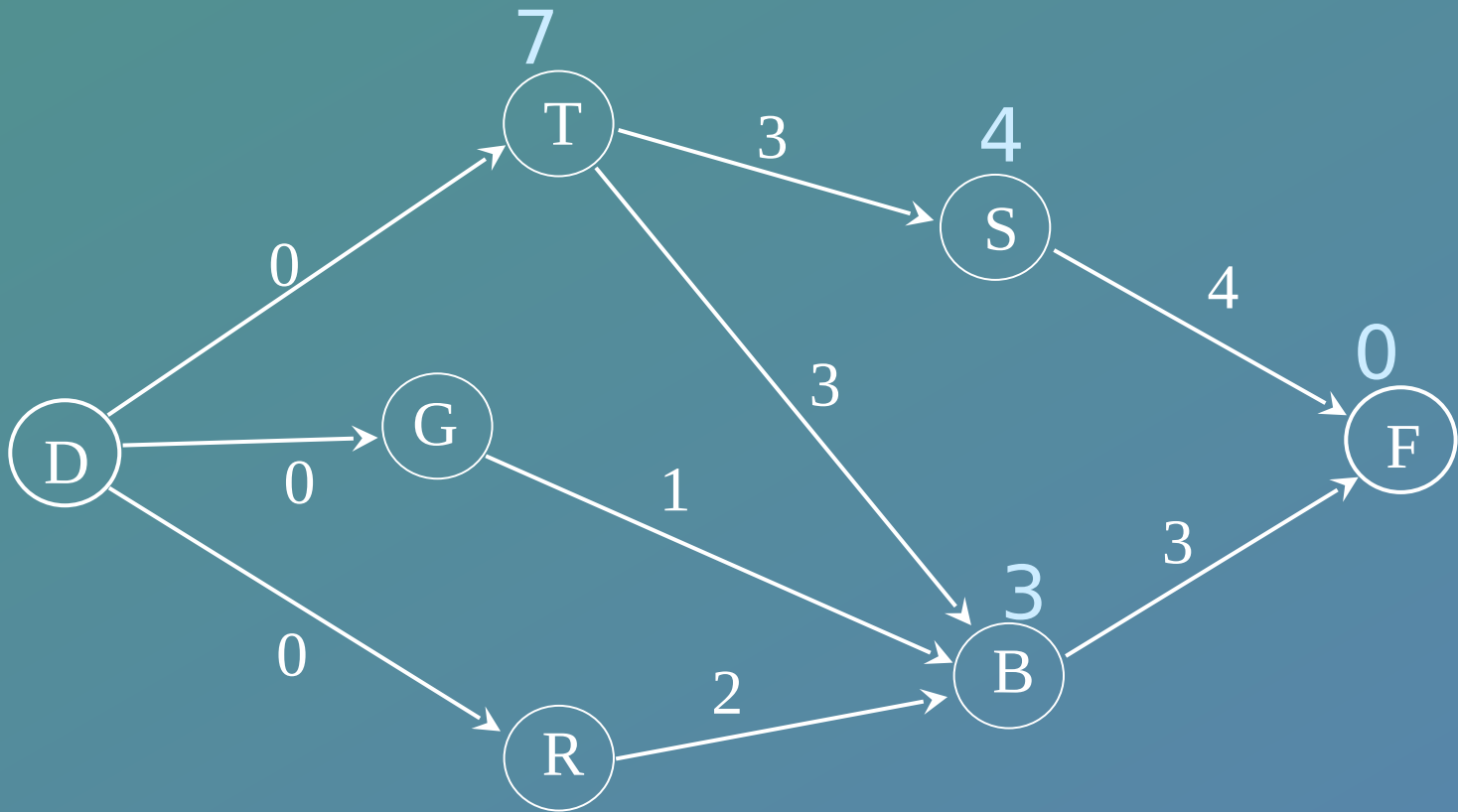
potentiels-tâches



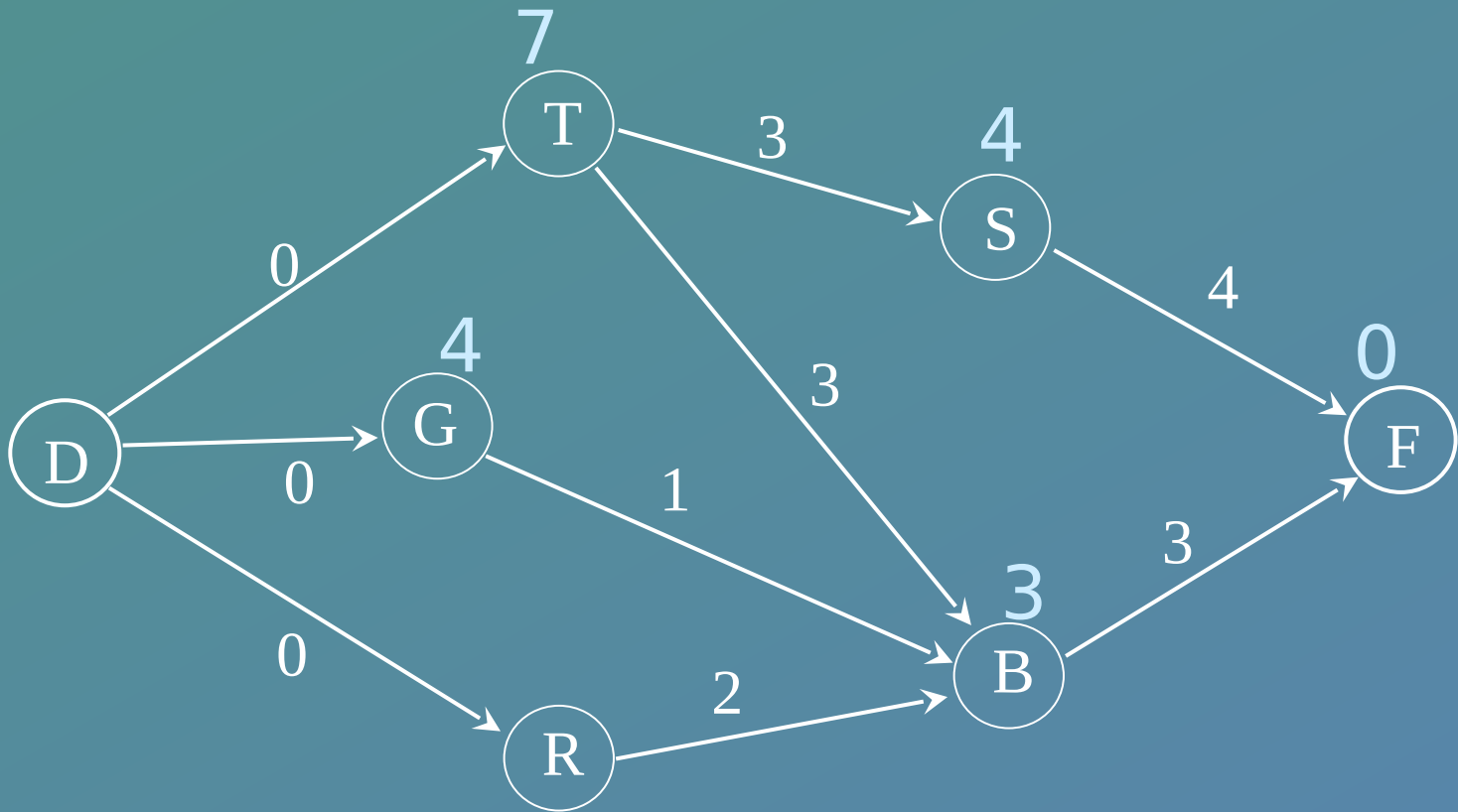
potentiels-tâches



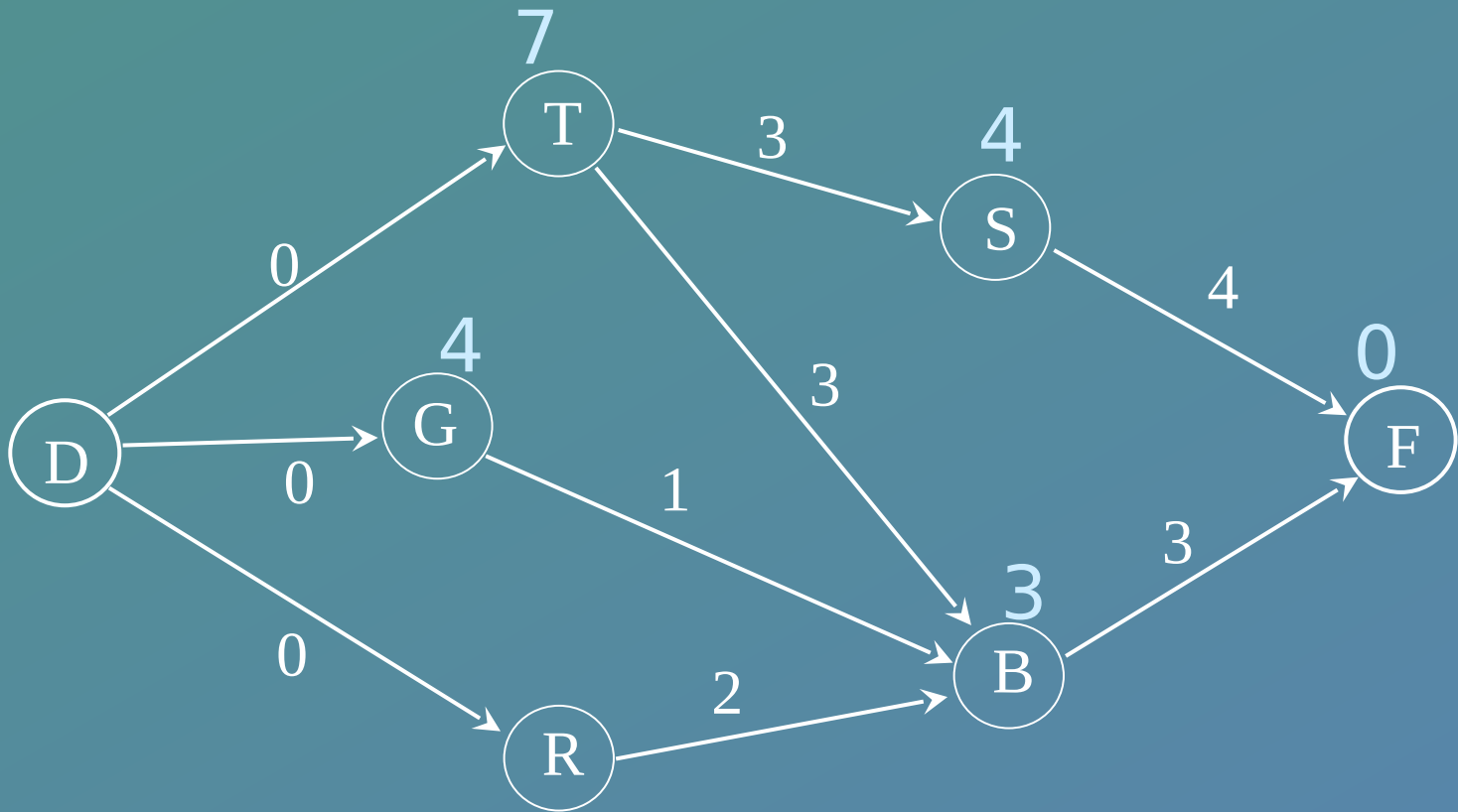
potentiels-tâches



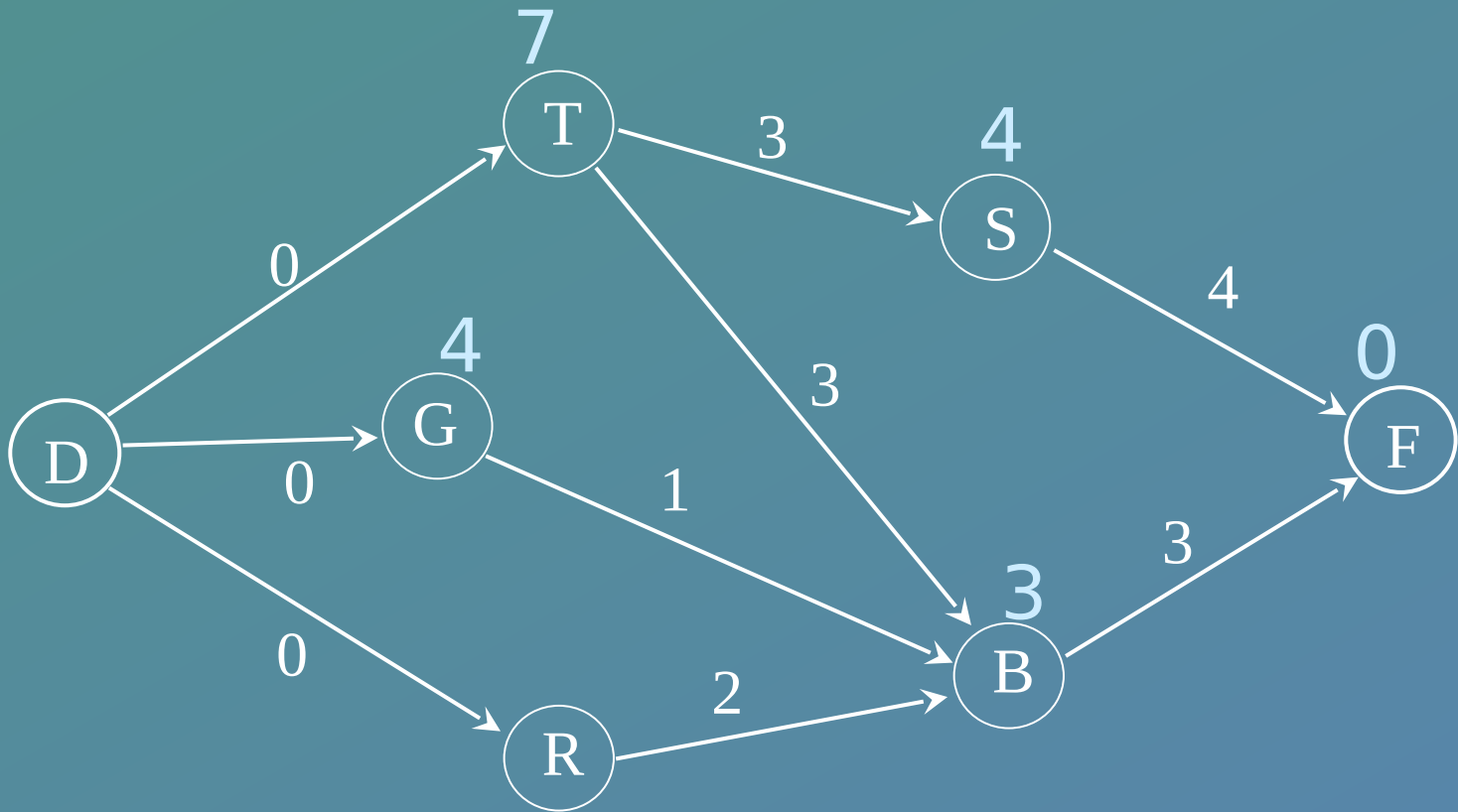
potentiels-tâches



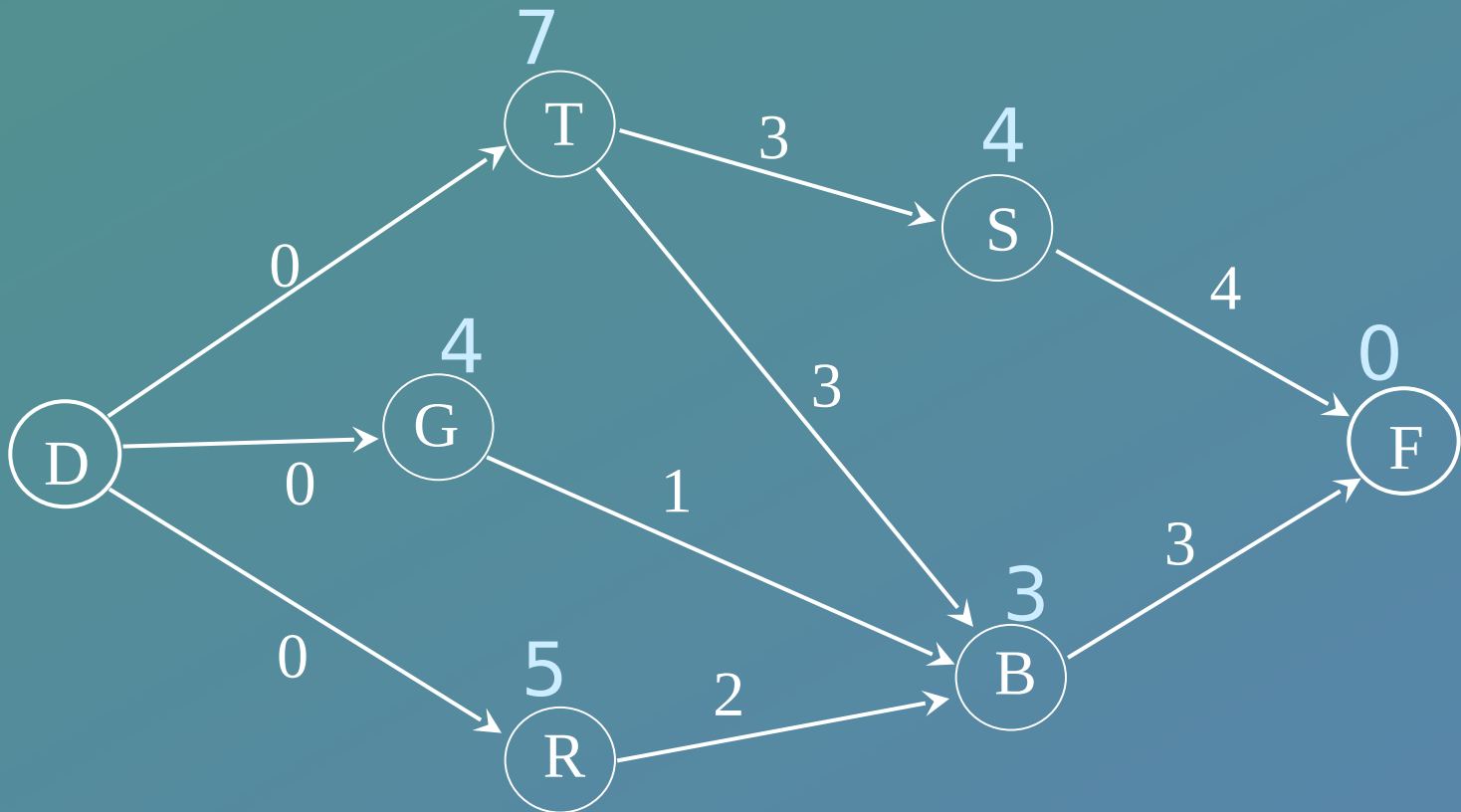
potentiels-tâches



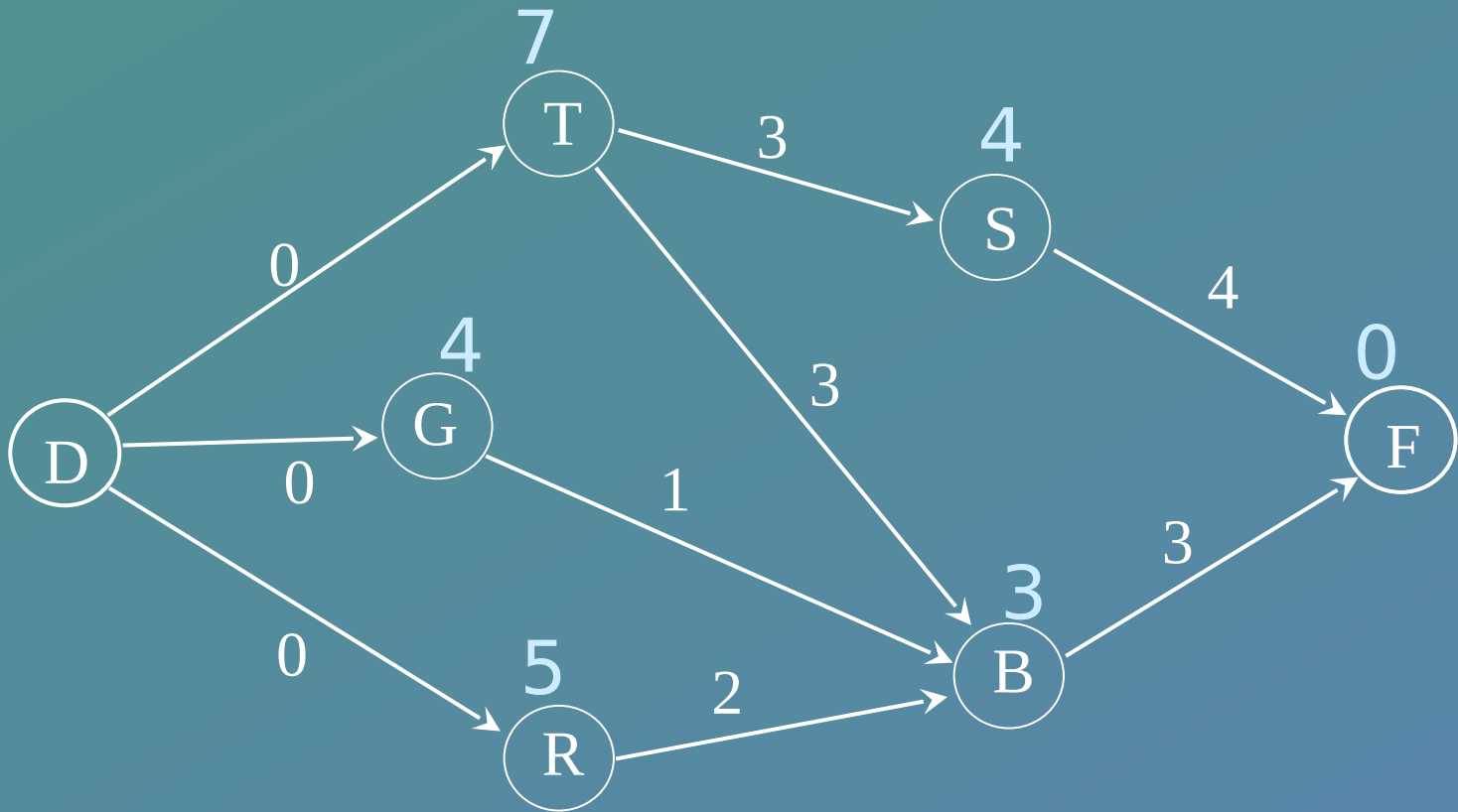
potentiels-tâches



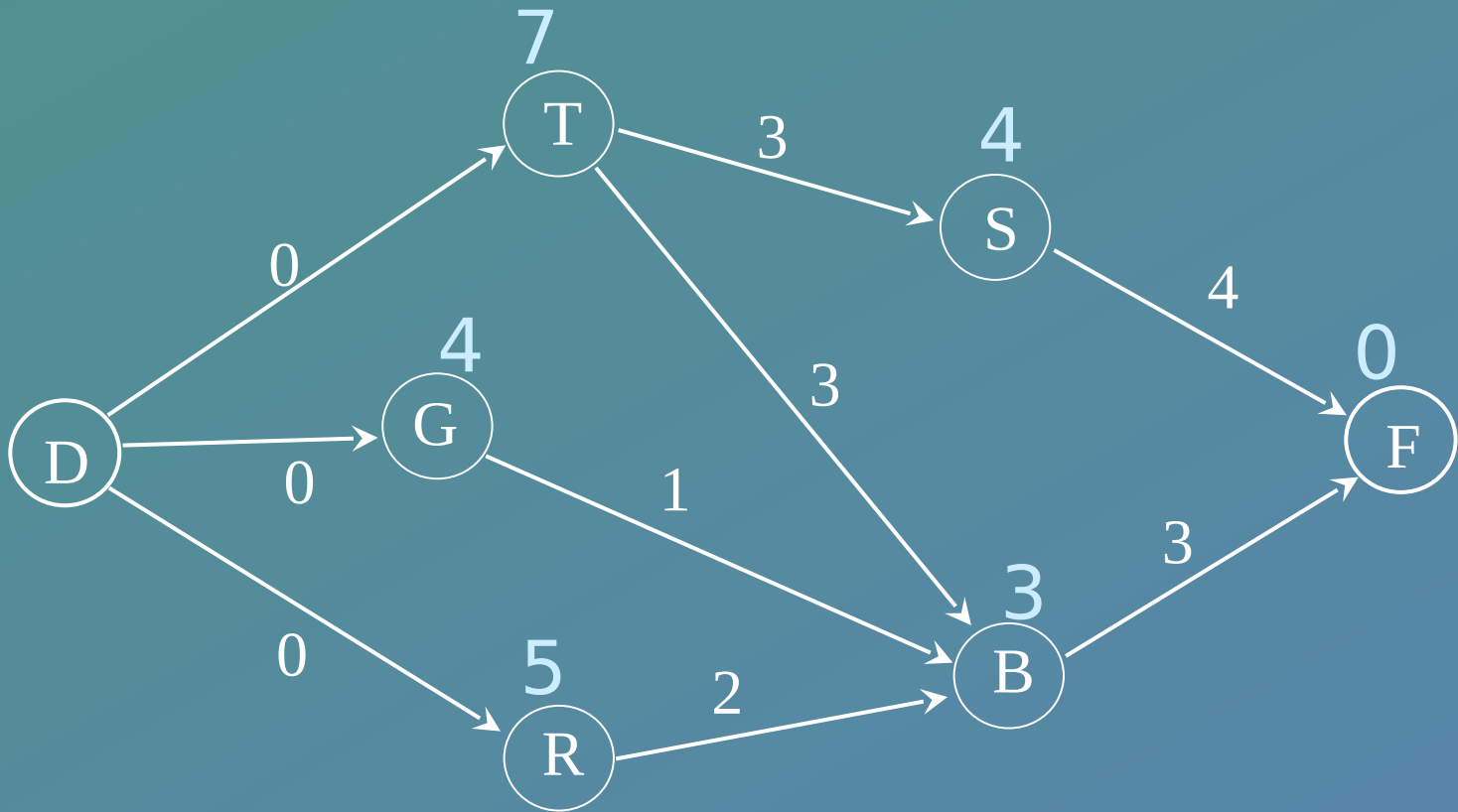
potentiels-tâches



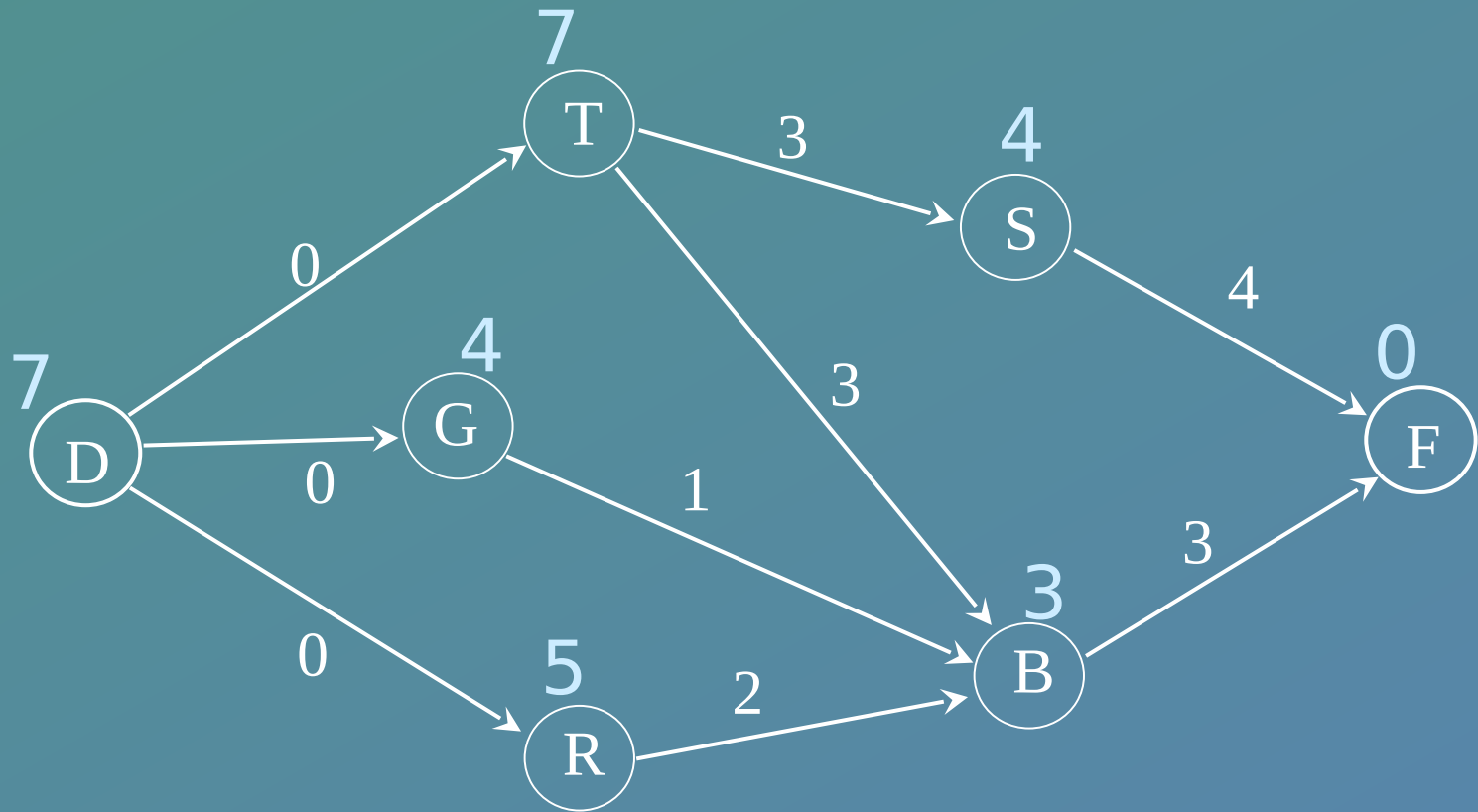
potentiels-tâches



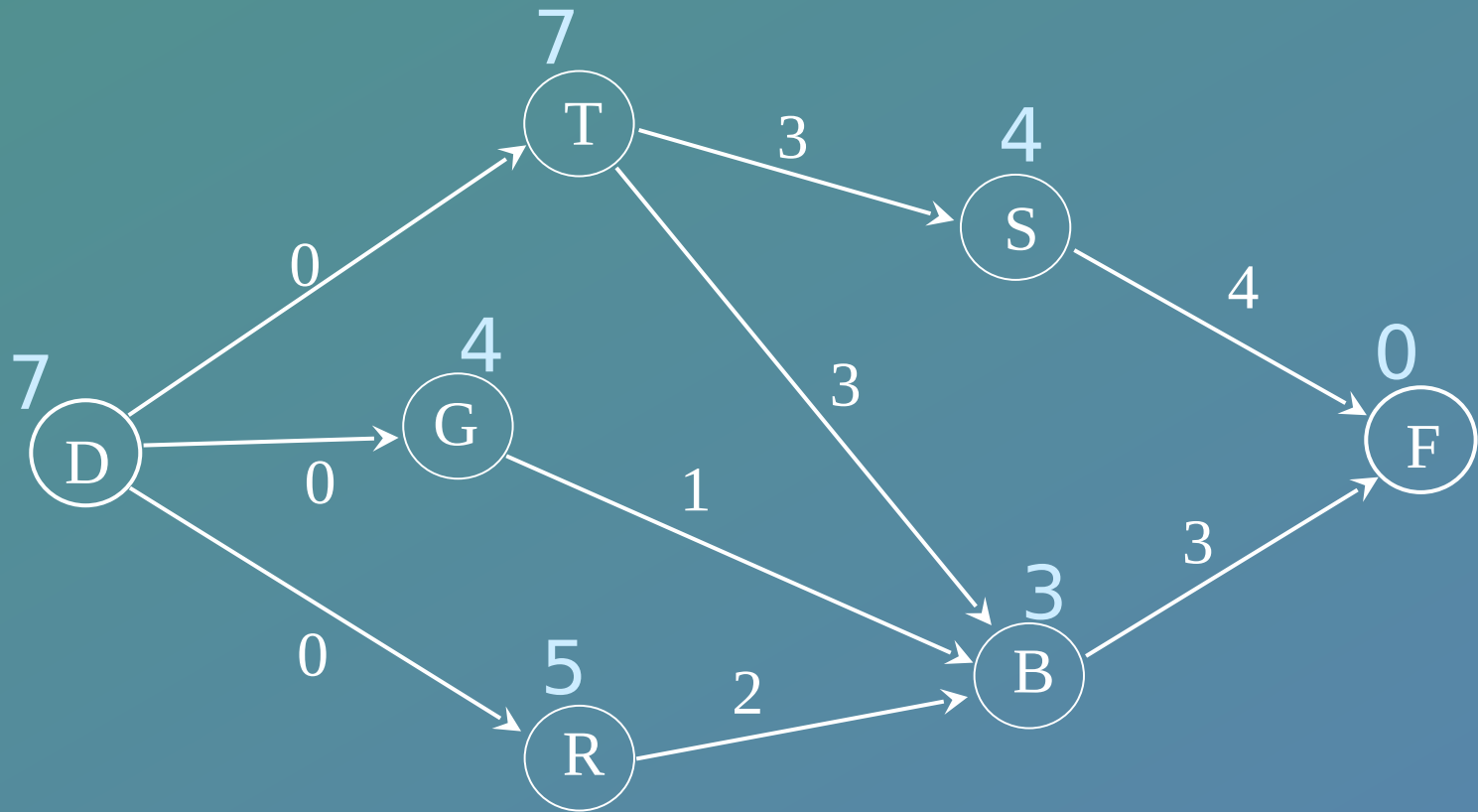
potentiels-tâches



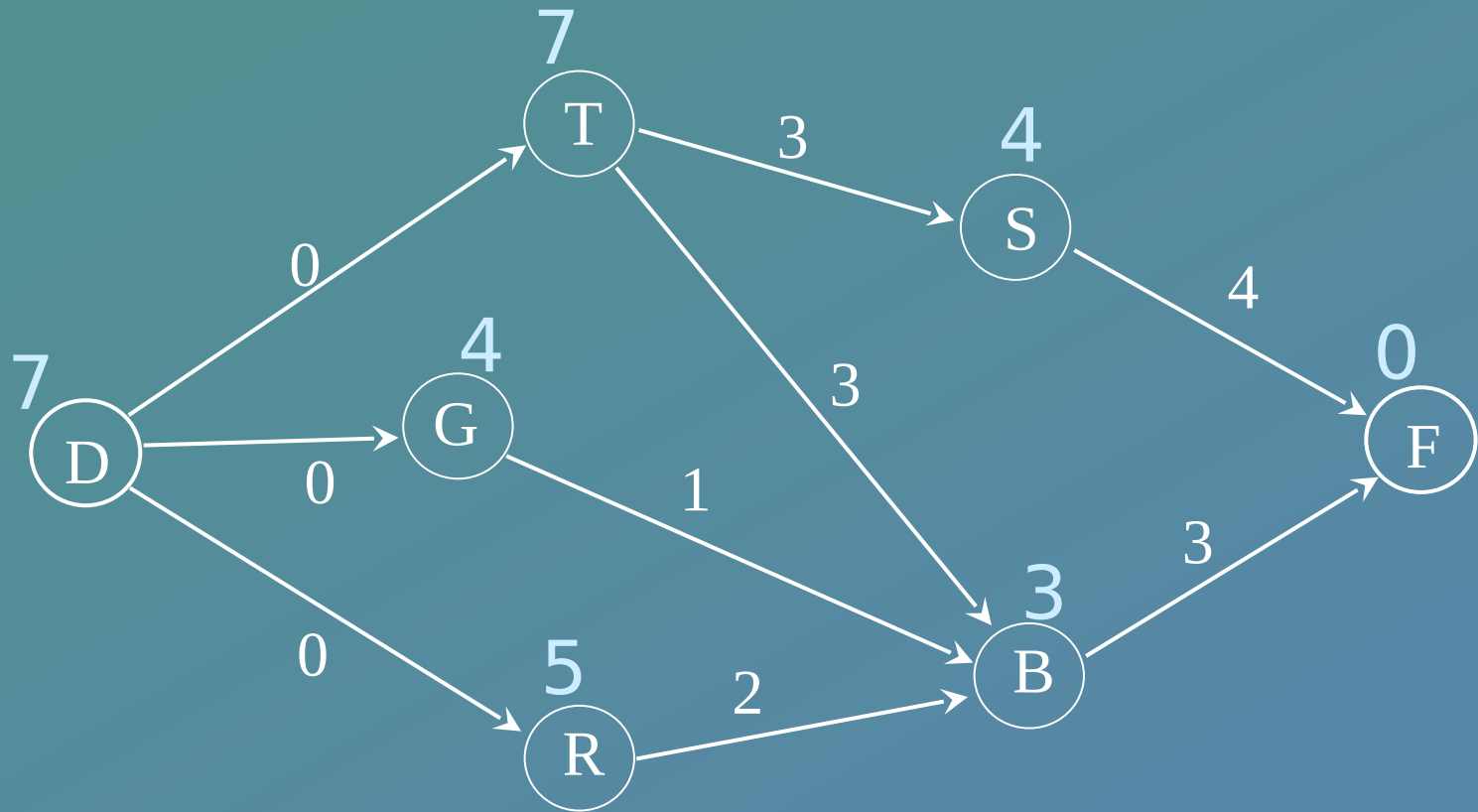
potentiels-tâches



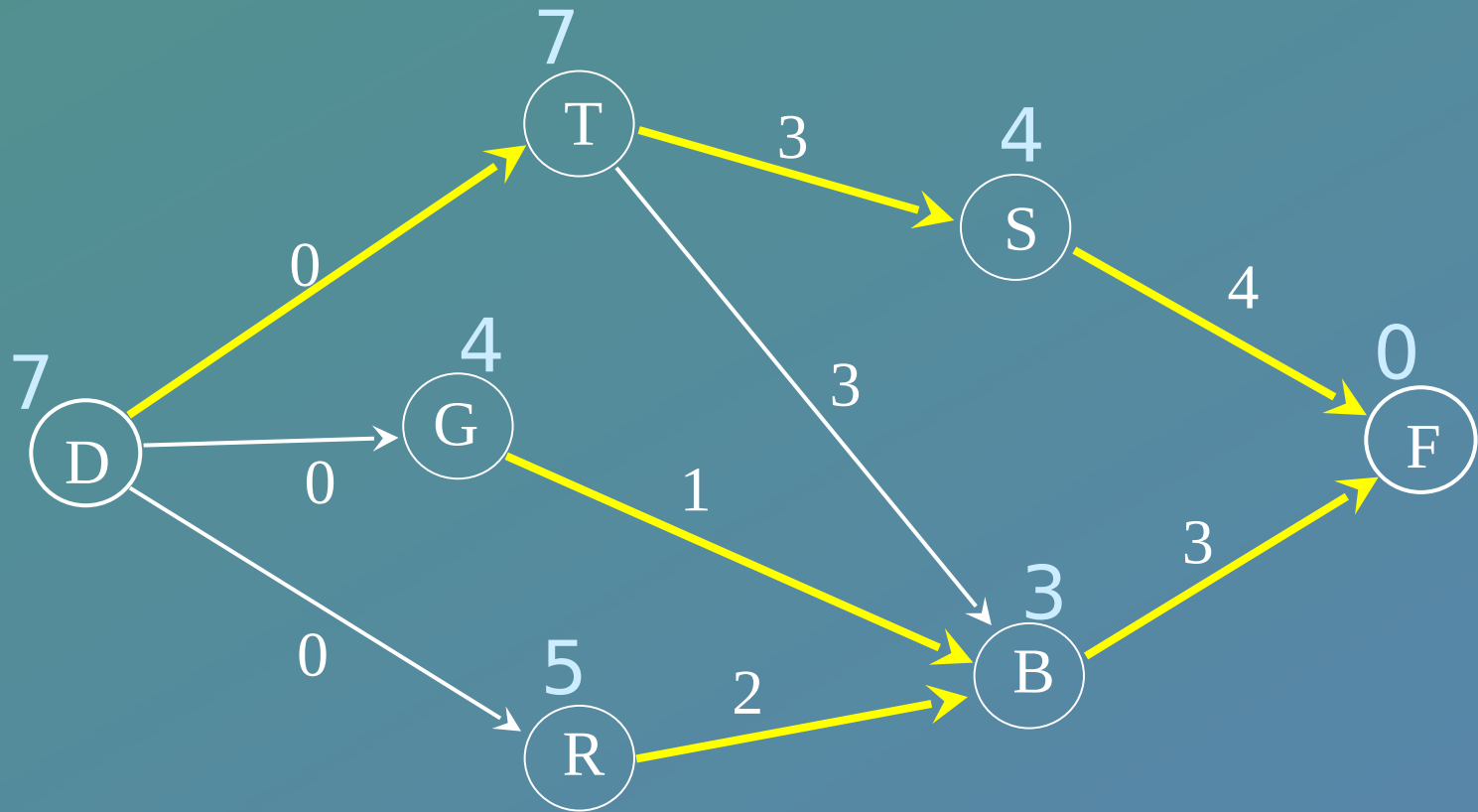
potentiels-tâches



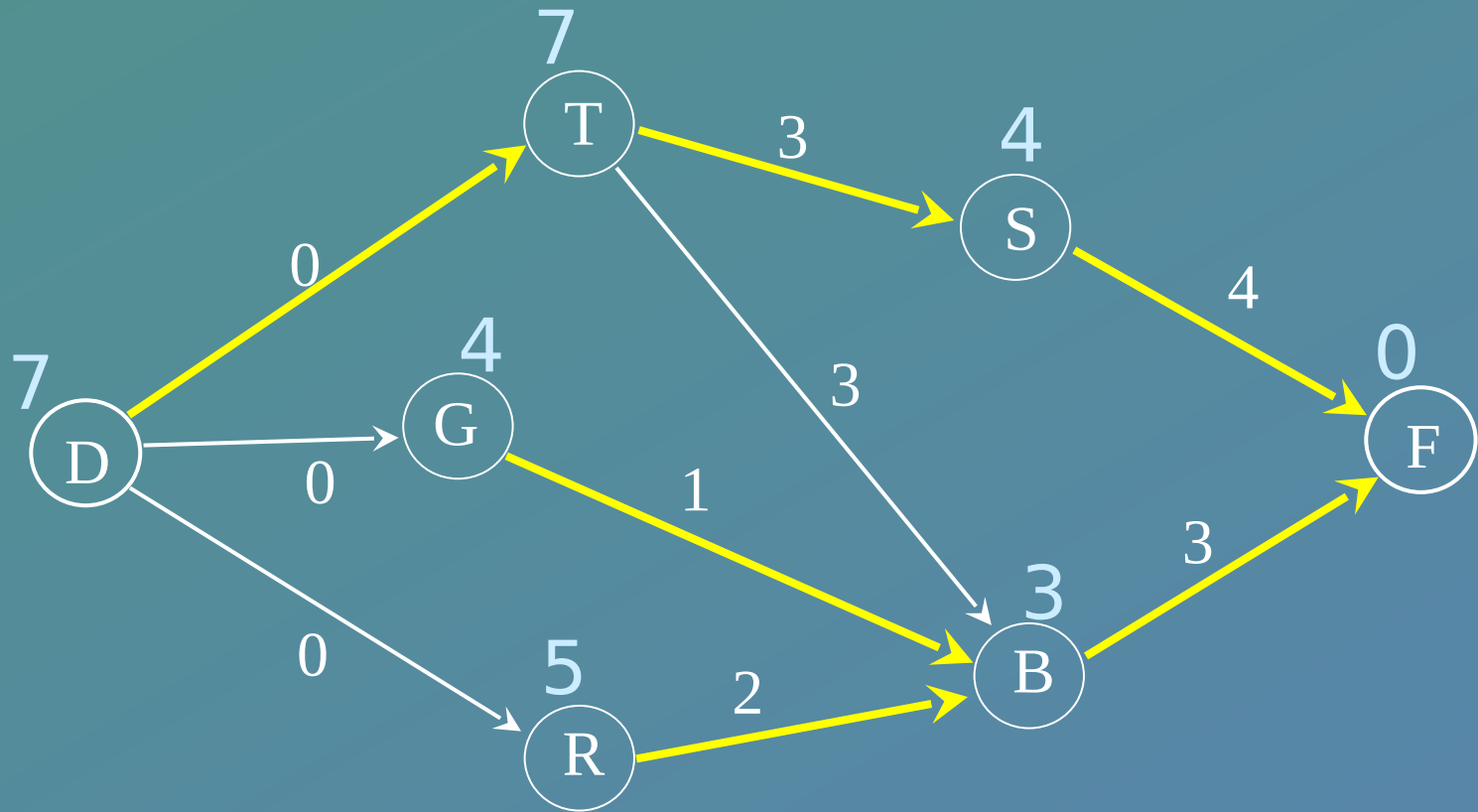
potentiels-tâches



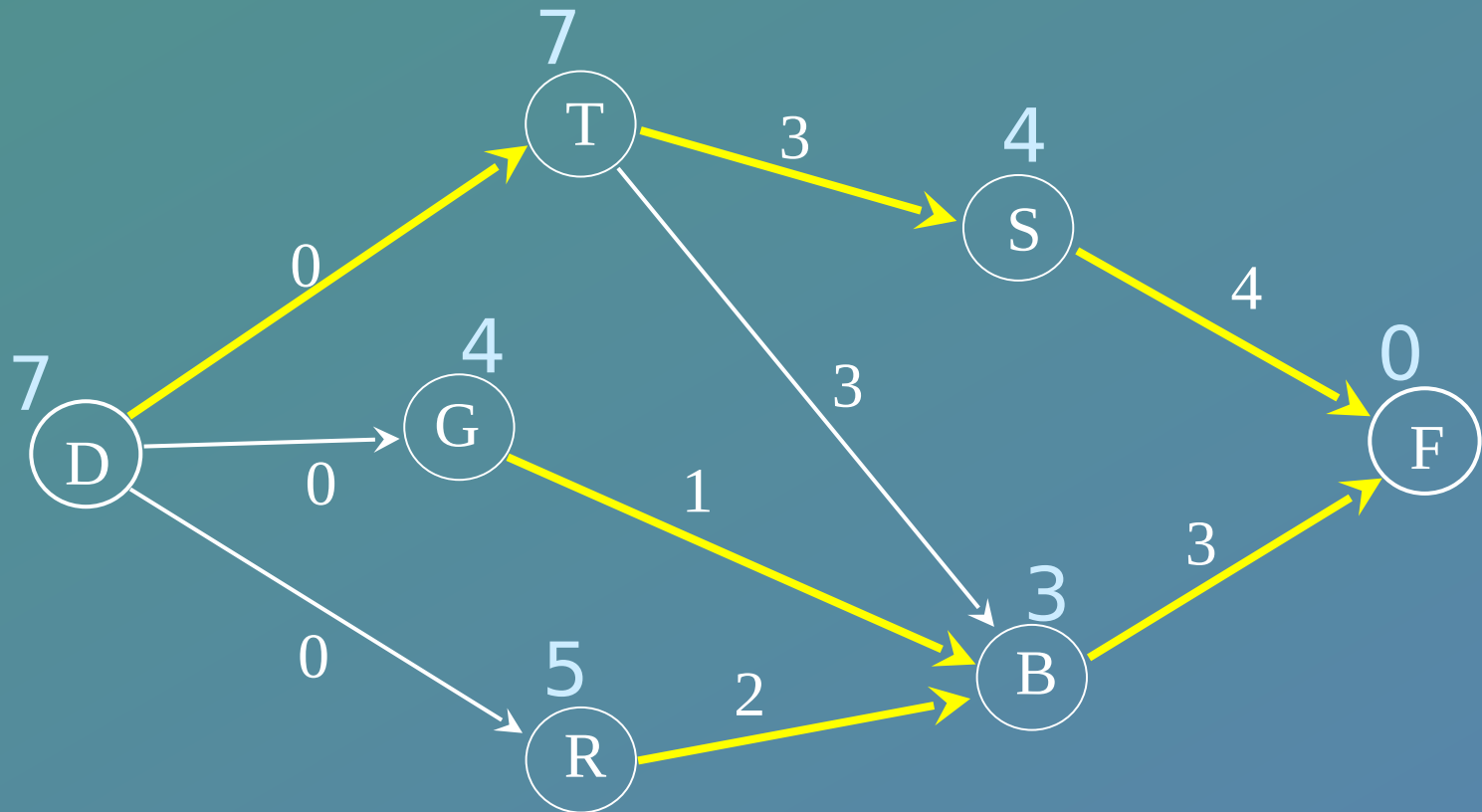
potentiels-tâches



potentiels-tâches



potentiels-tâches

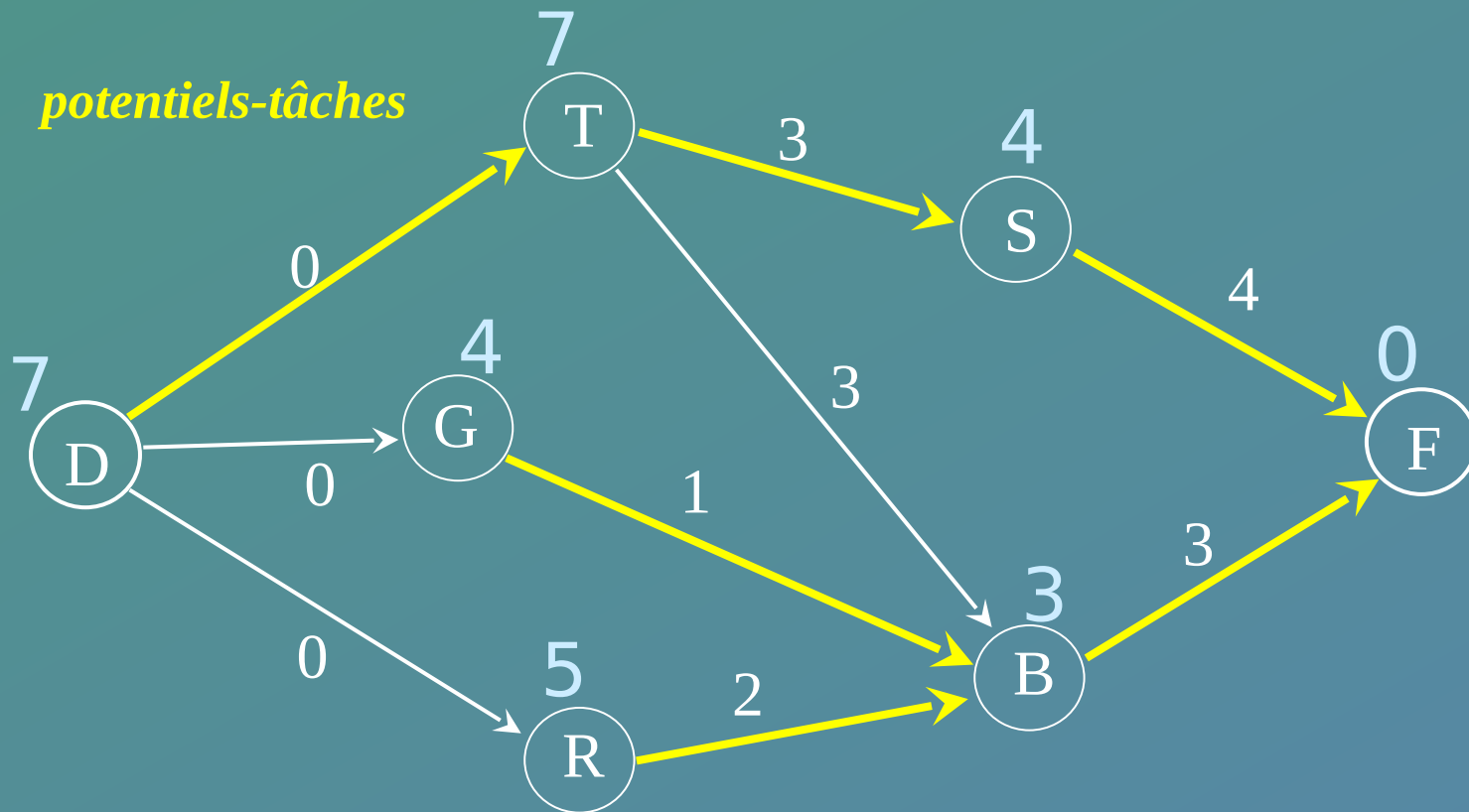


Les dates au plus tard sont calculées grâce à la relation :

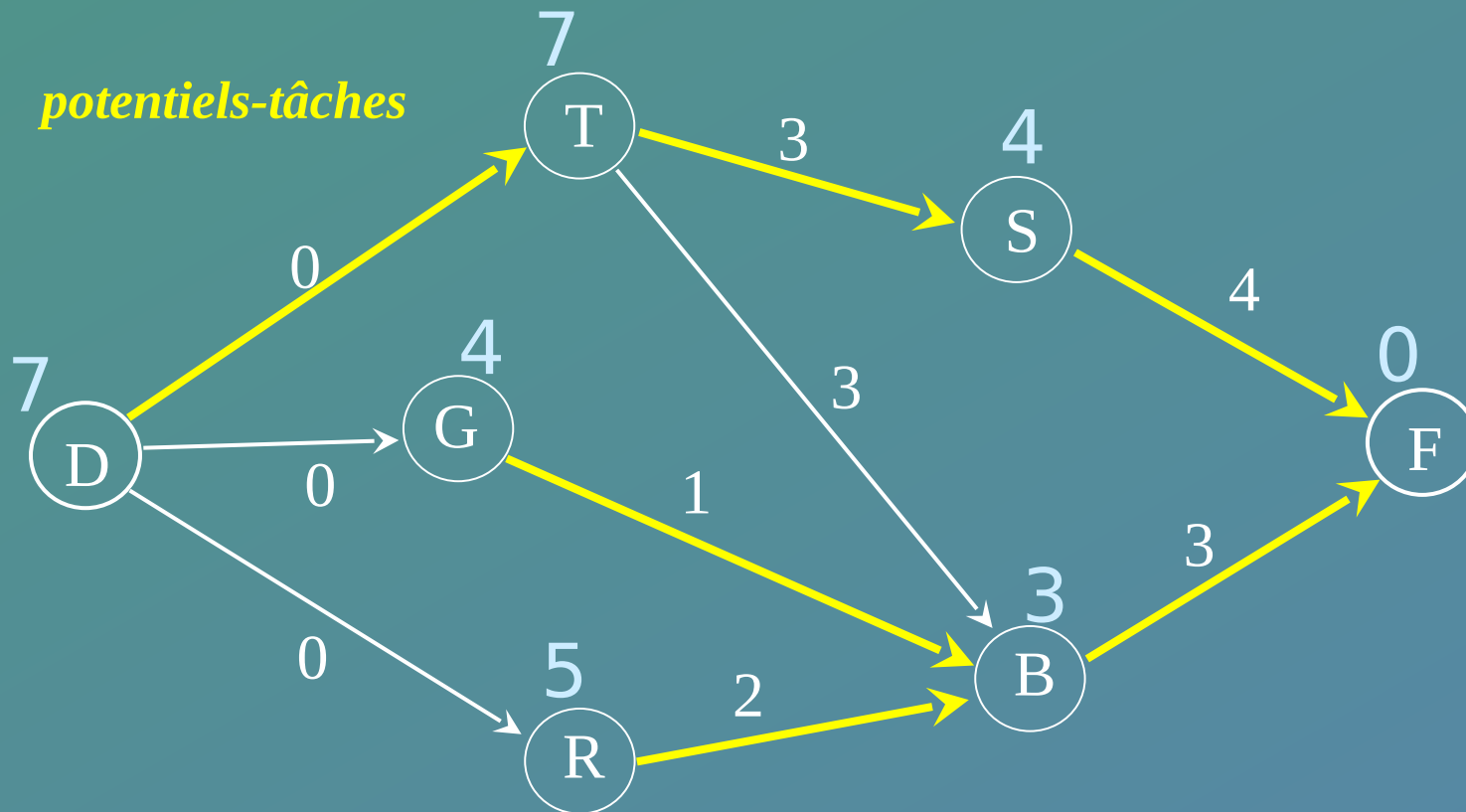
$$\eta(x) = \pi(F) - \pi'(x) .$$



potentiels-tâches



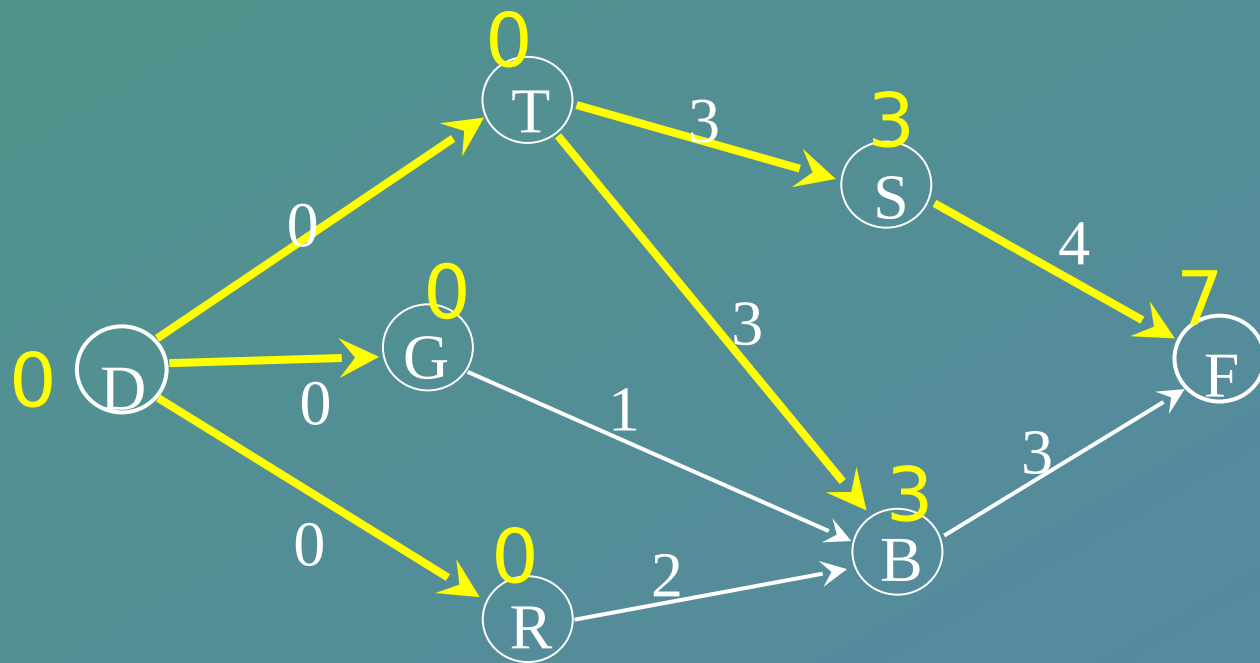
potentiels-tâches



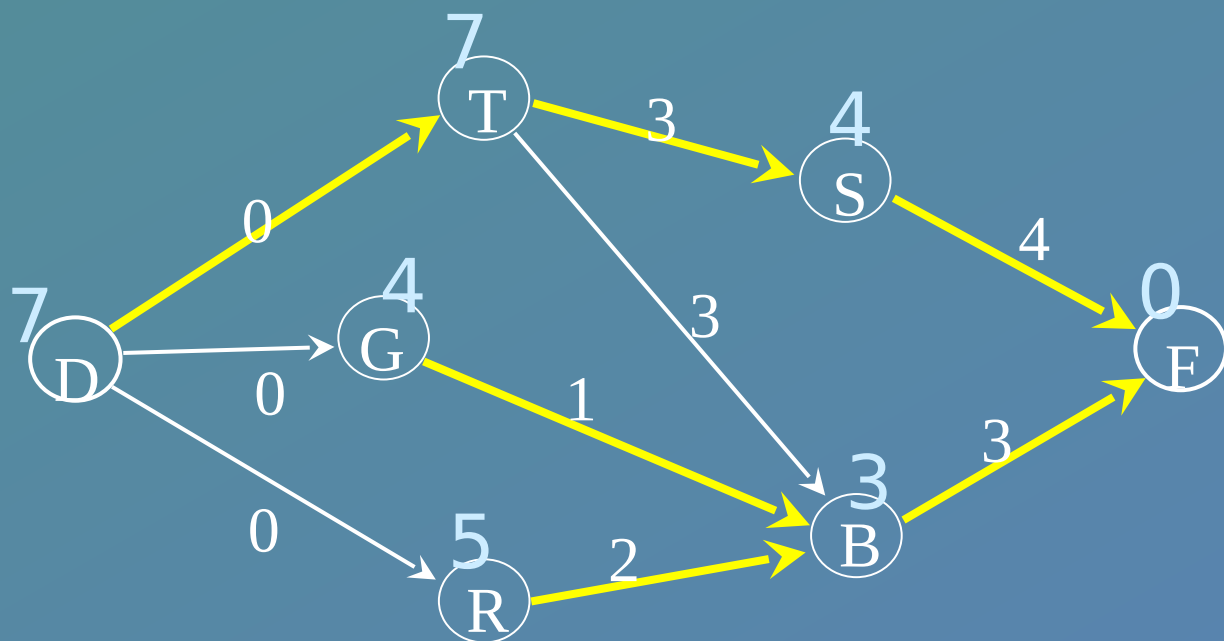
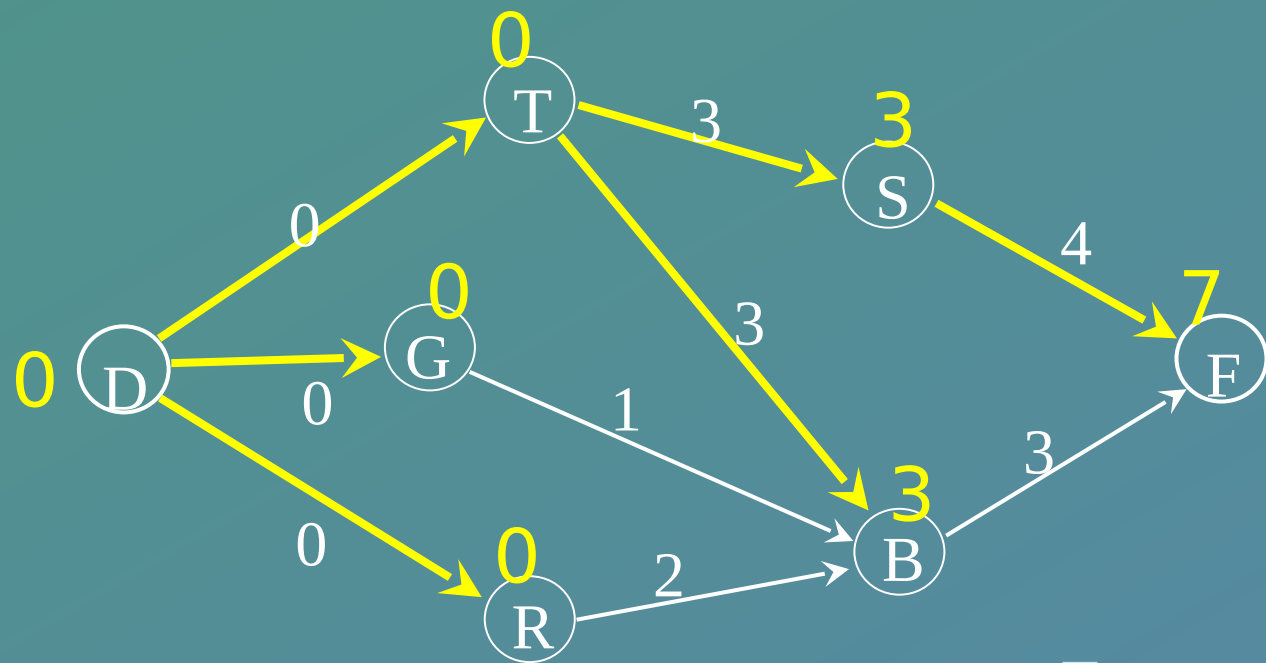
tâche	durée	après :	la date au plus tard
Début	0	-	0
T	3	Début	0
G	1	Début	3
R	2	Début	2
B	3	T, G, R	4
S	4	T	3
Fin	0	T, G, R, B, S	7



potentiels-tâches



potentiels-tâches





potentiels-événements



potentiels-événements

P E R T



potentiels-événements

P E R T

Program Evaluation and Review Technique



potentiels-événements

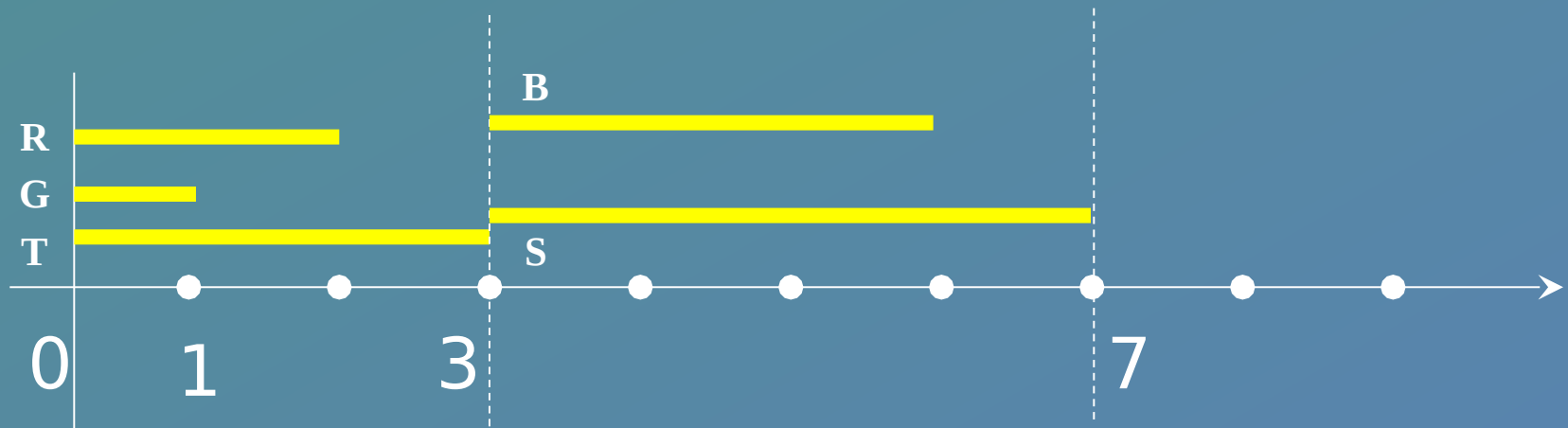
P E R T

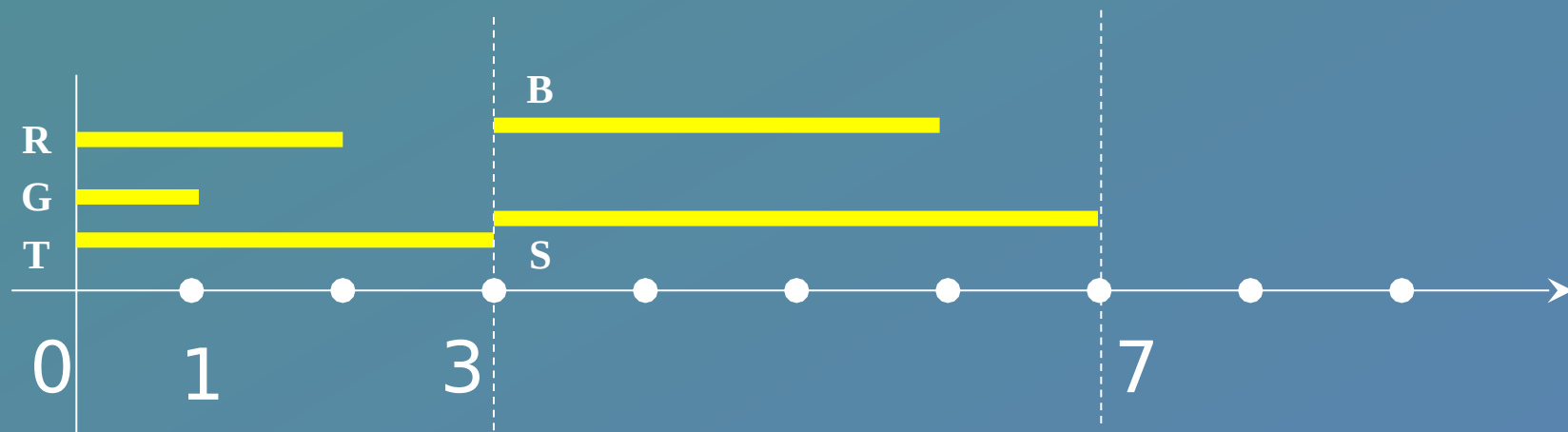
Program Evaluation and Review Technique

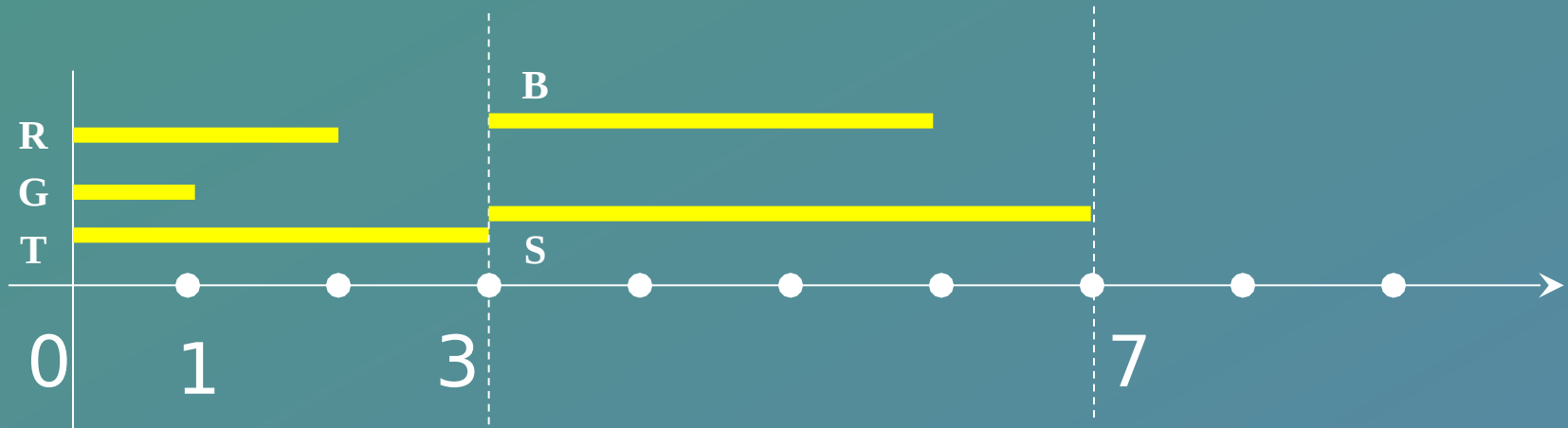
Pour Eviter les Retards Traditionnels



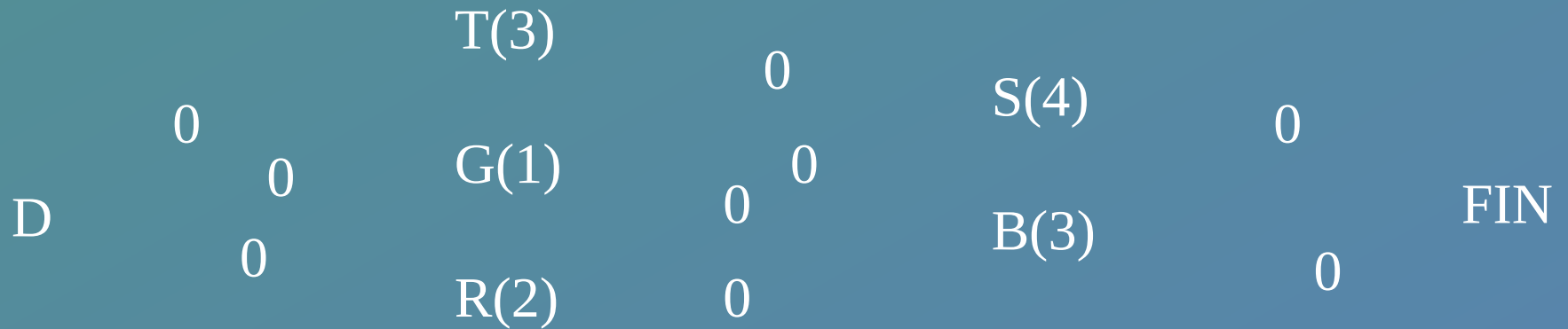
tâche	durée (jours)	contraintes ; après :
T	3	—
G	1	—
R	2	—
B	3	T, G, R
S	4	T

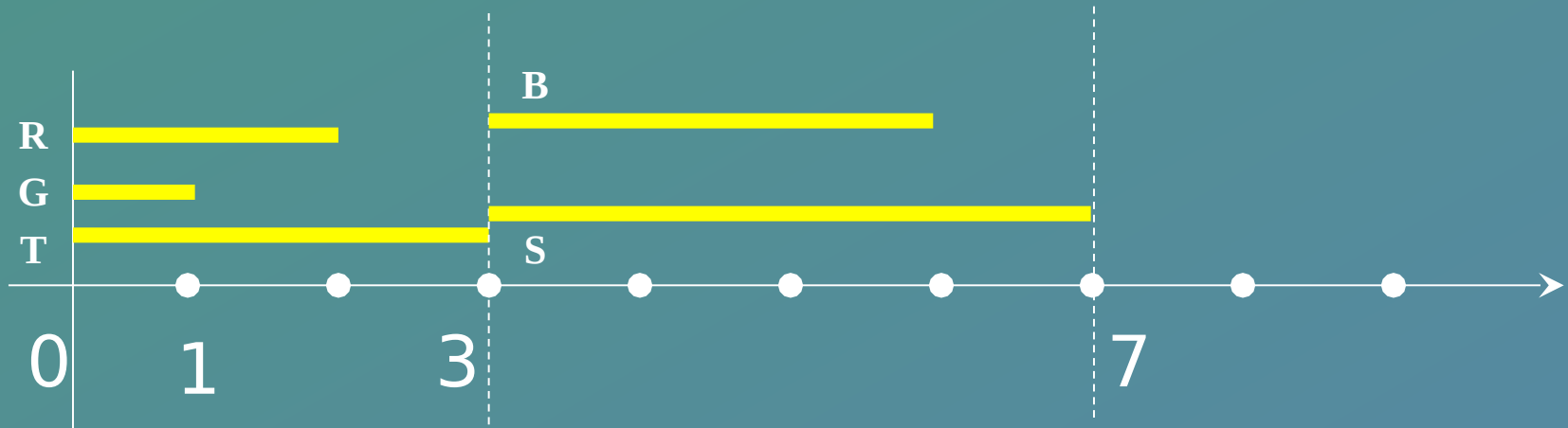




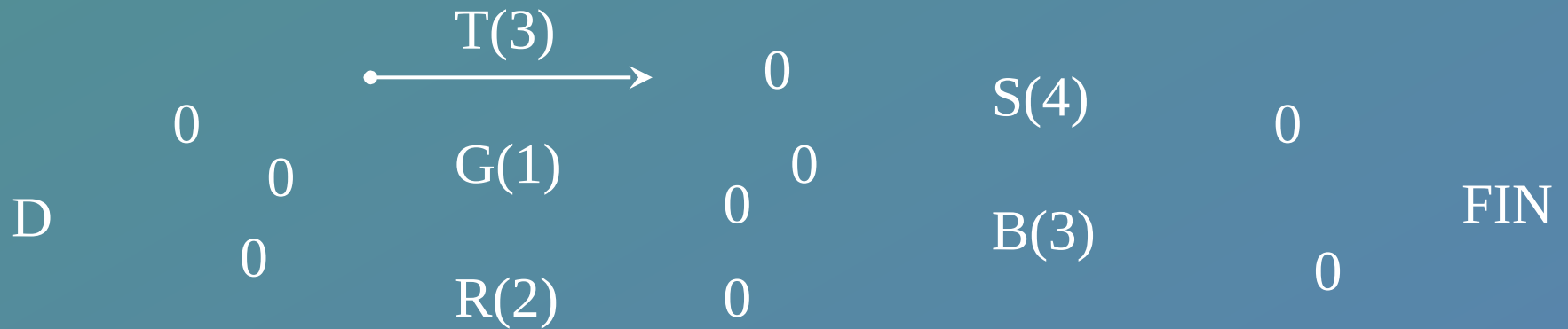


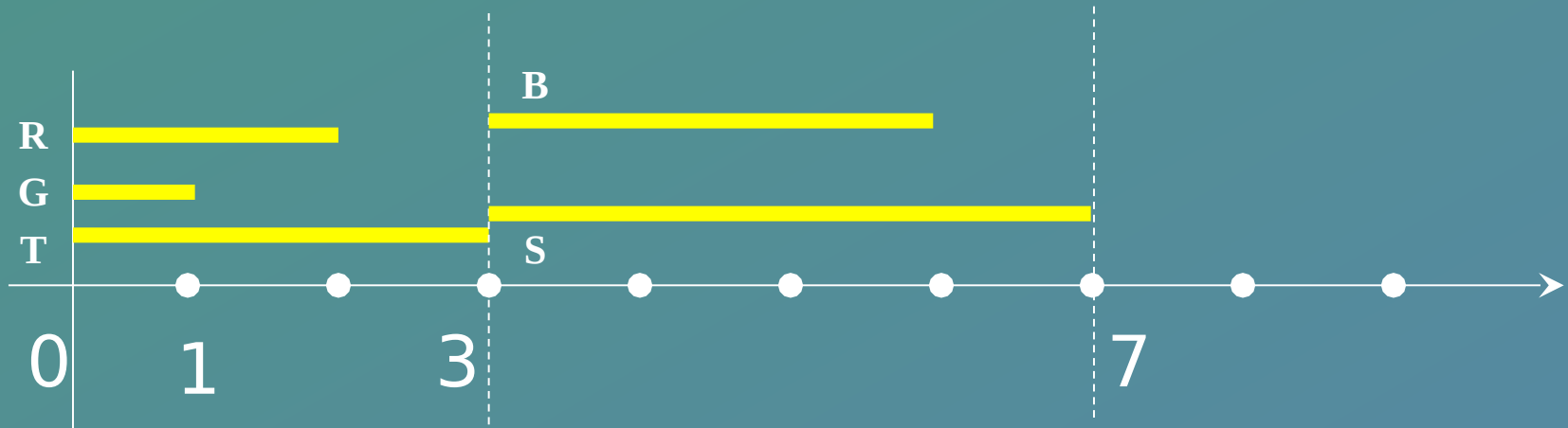
modélisation PERT : une tâche est représentée par un arc



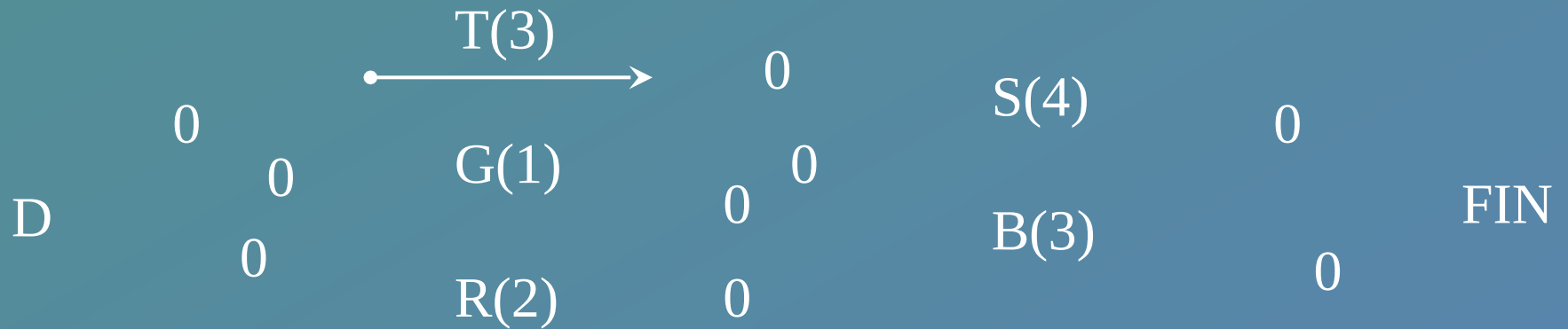


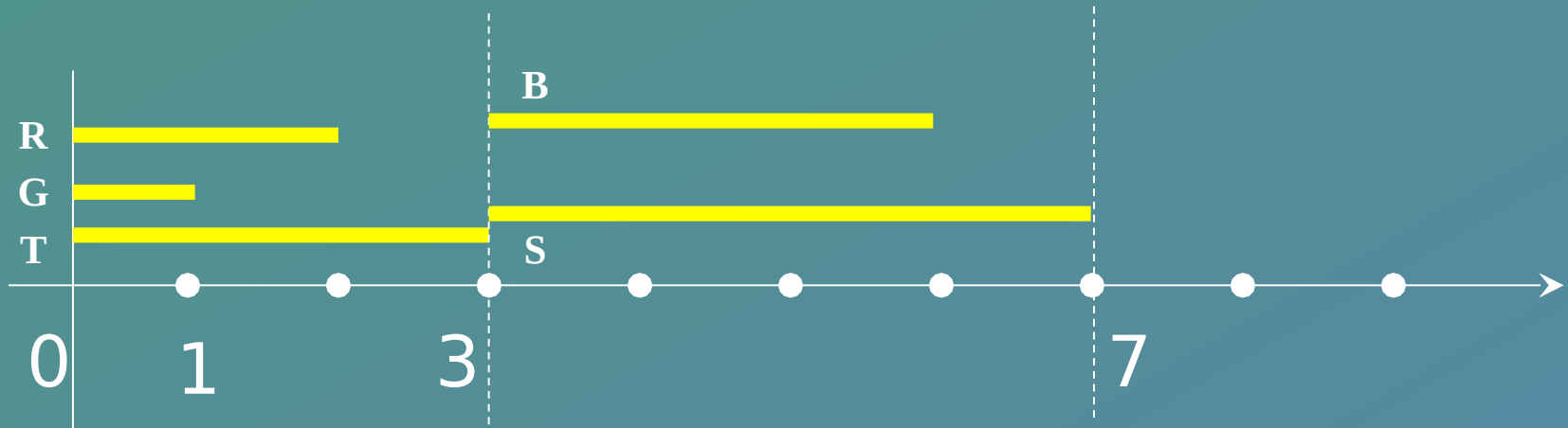
modélisation PERT : une tâche est représentée par un arc



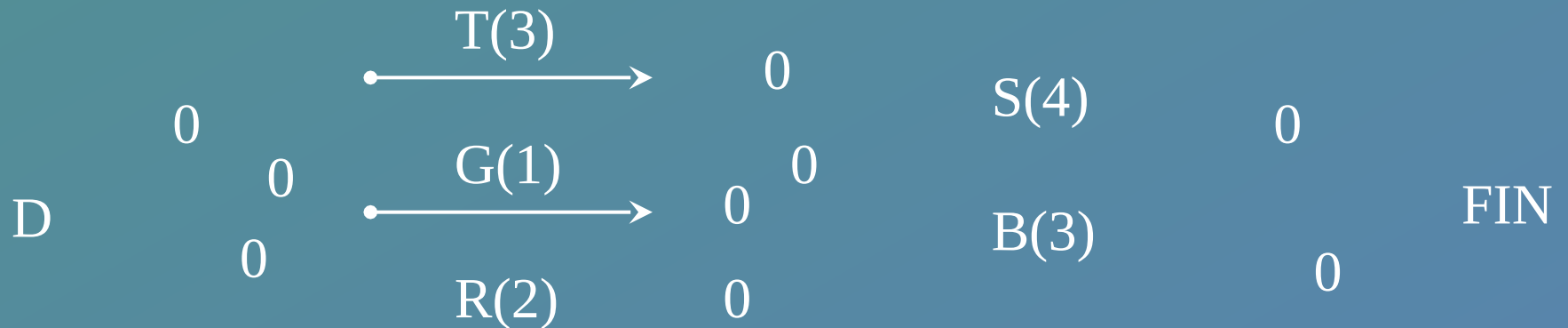


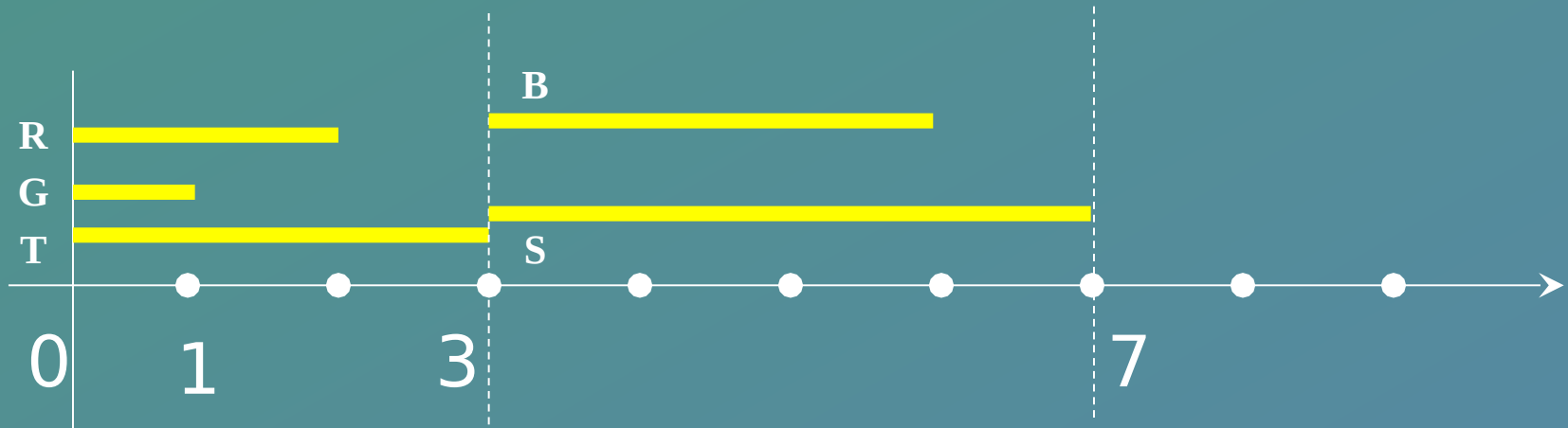
modélisation PERT : une tâche est représentée par un arc



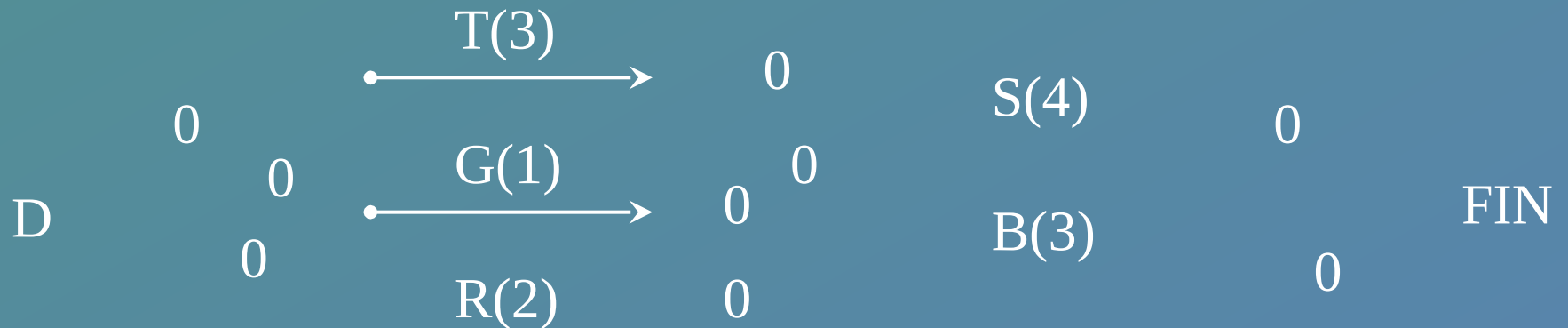


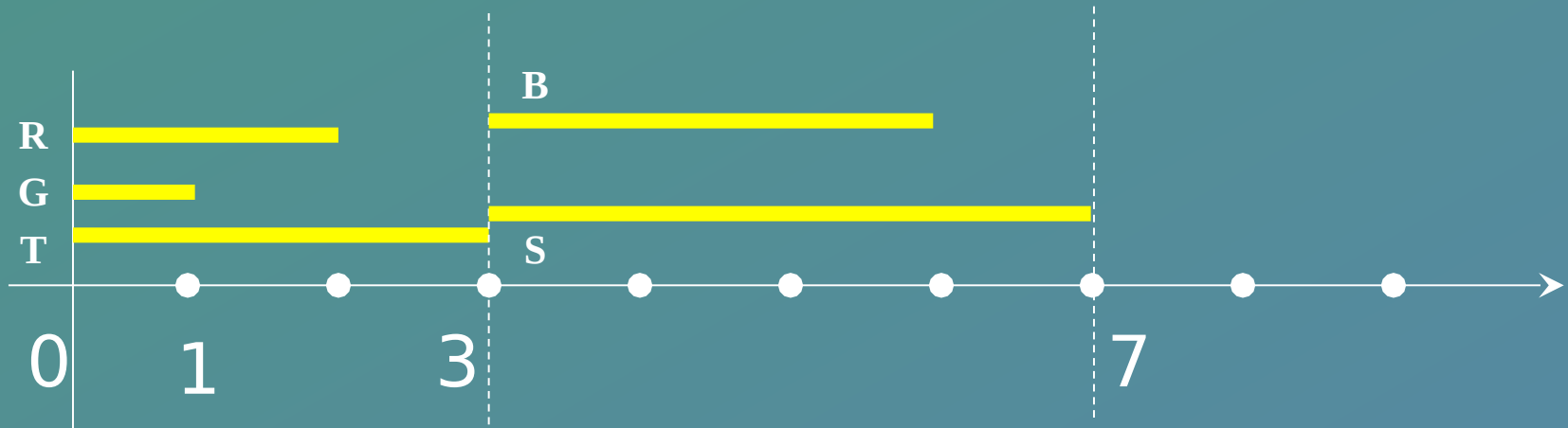
modélisation PERT : une tâche est représentée par un arc



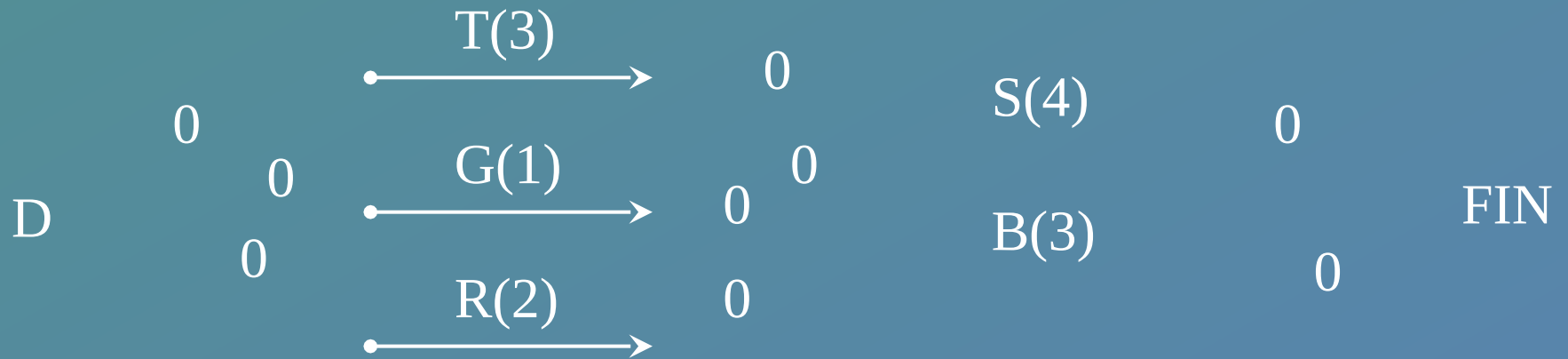


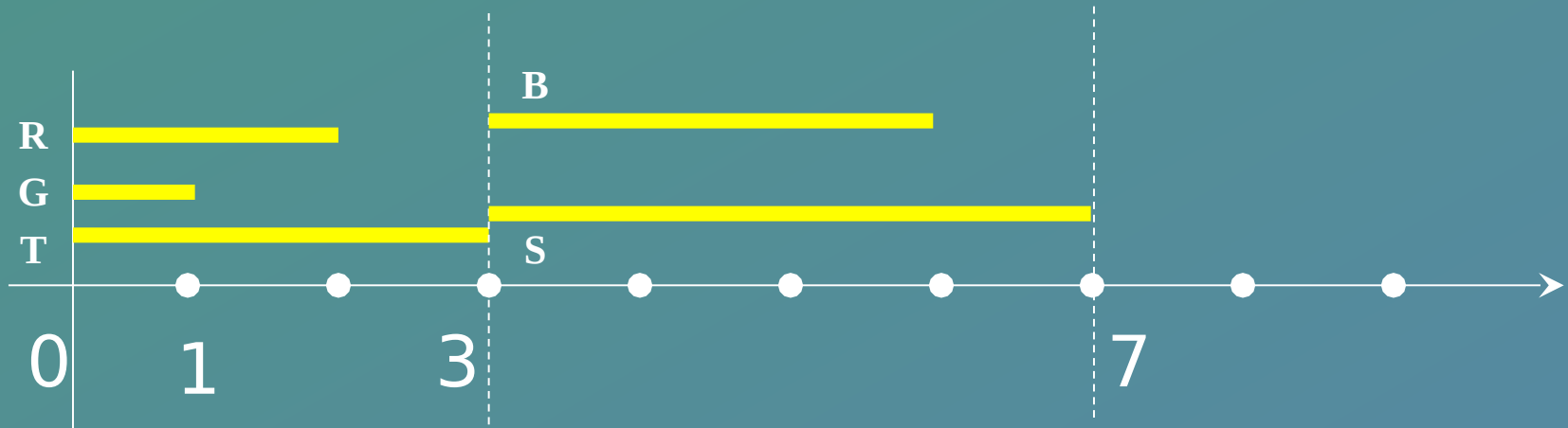
modélisation PERT : une tâche est représentée par un arc



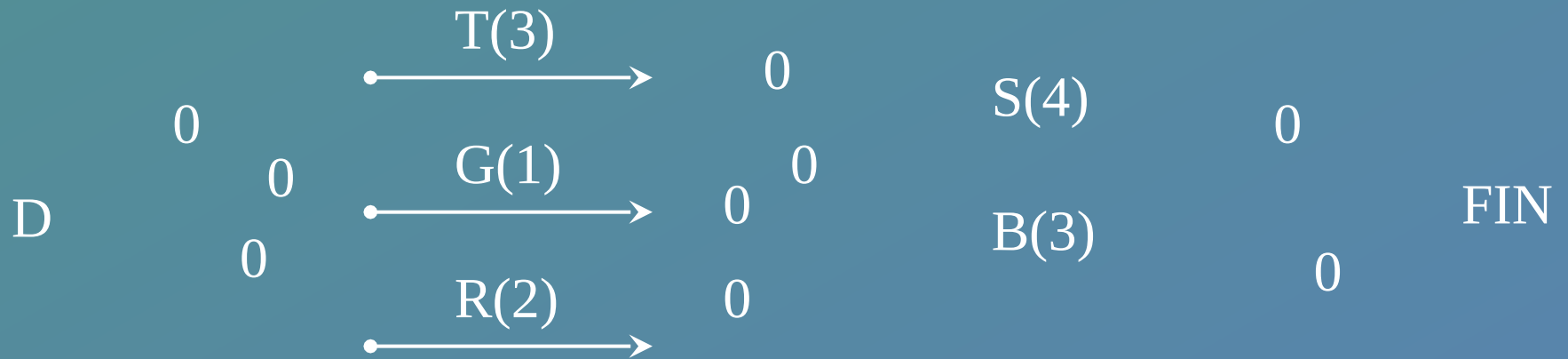


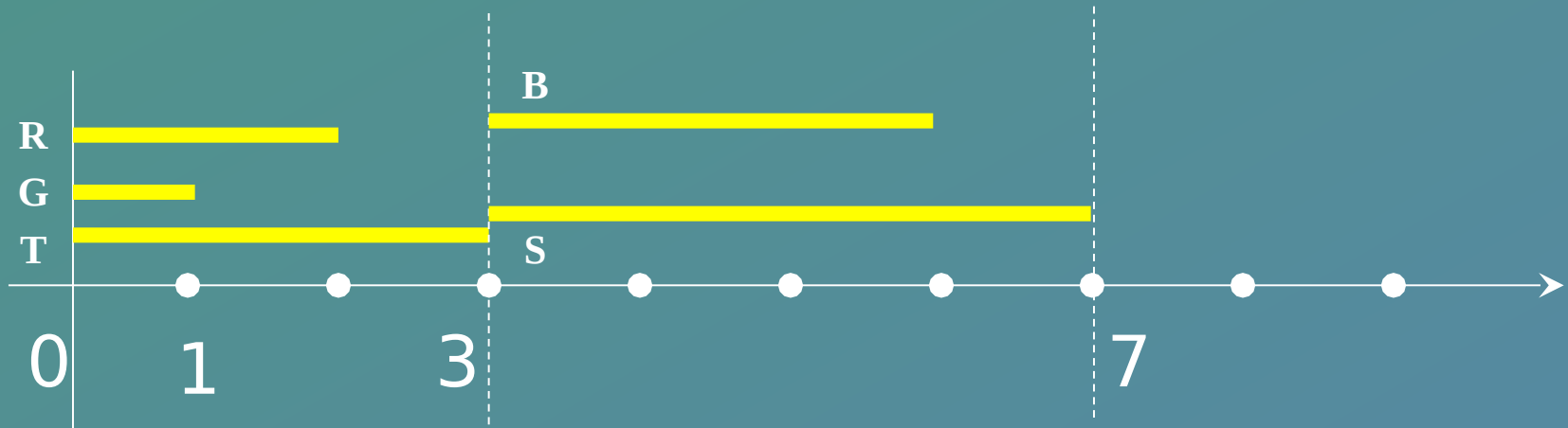
modélisation PERT : une tâche est représentée par un arc



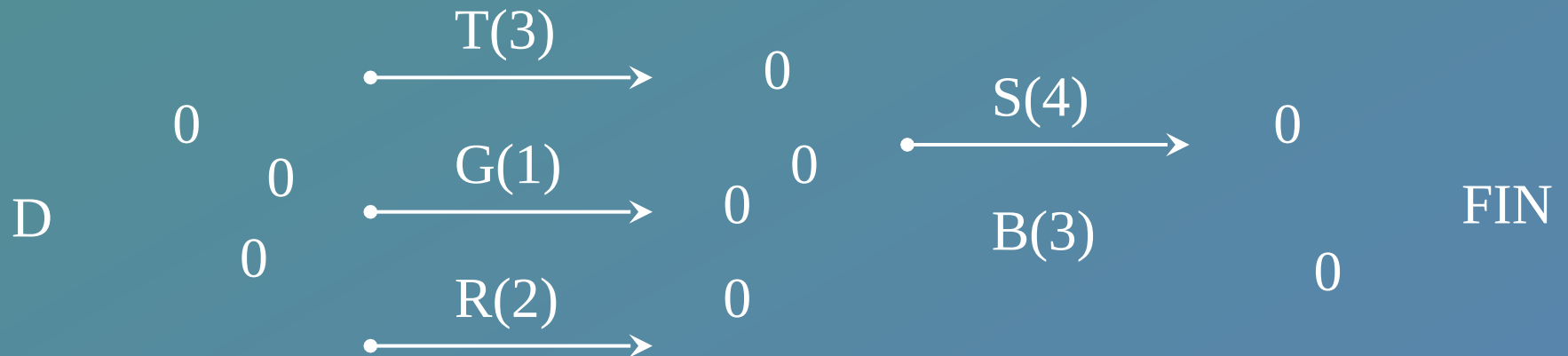


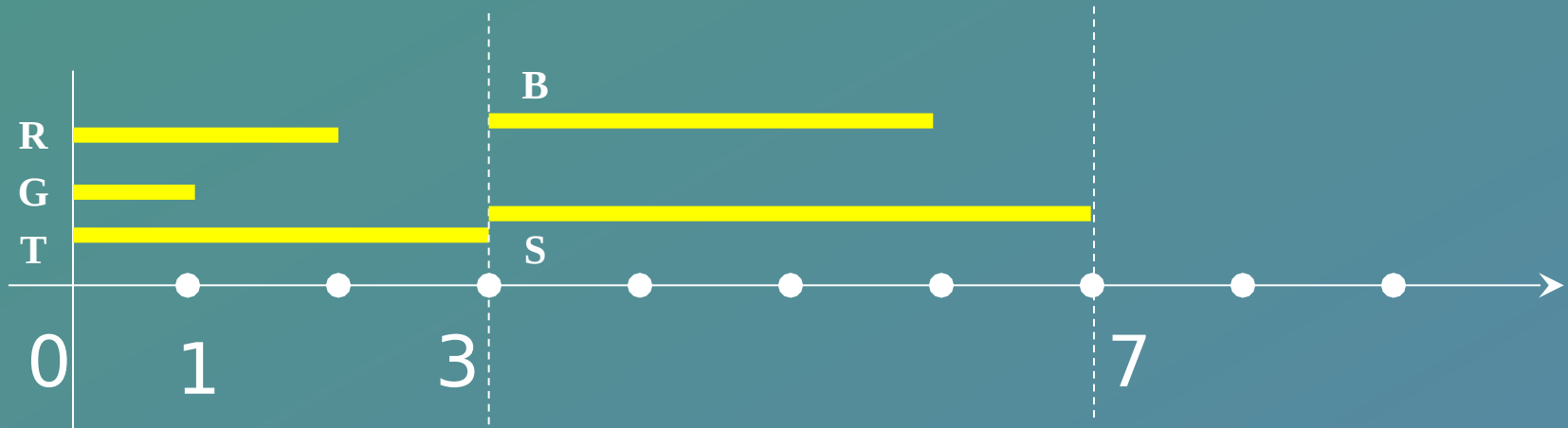
modélisation PERT : une tâche est représentée par un arc



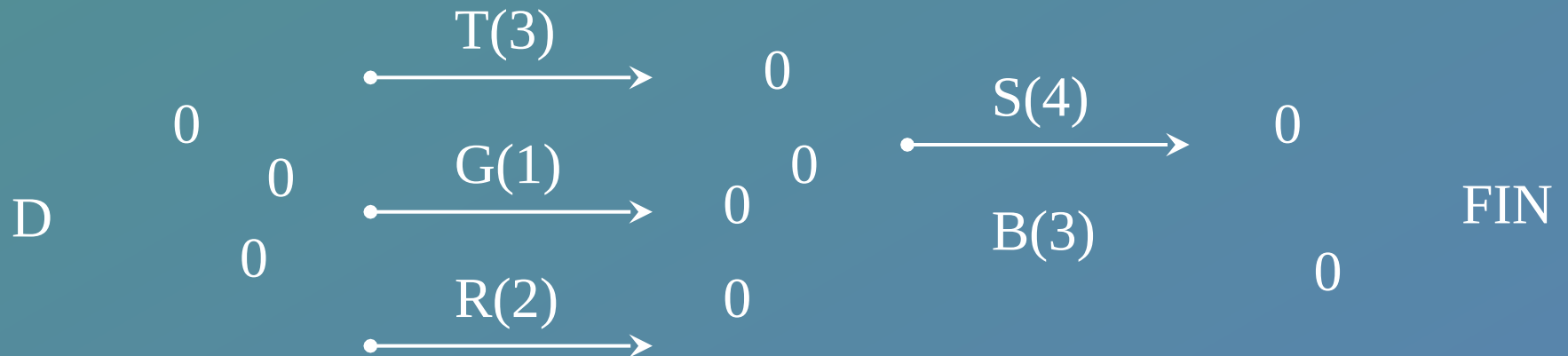


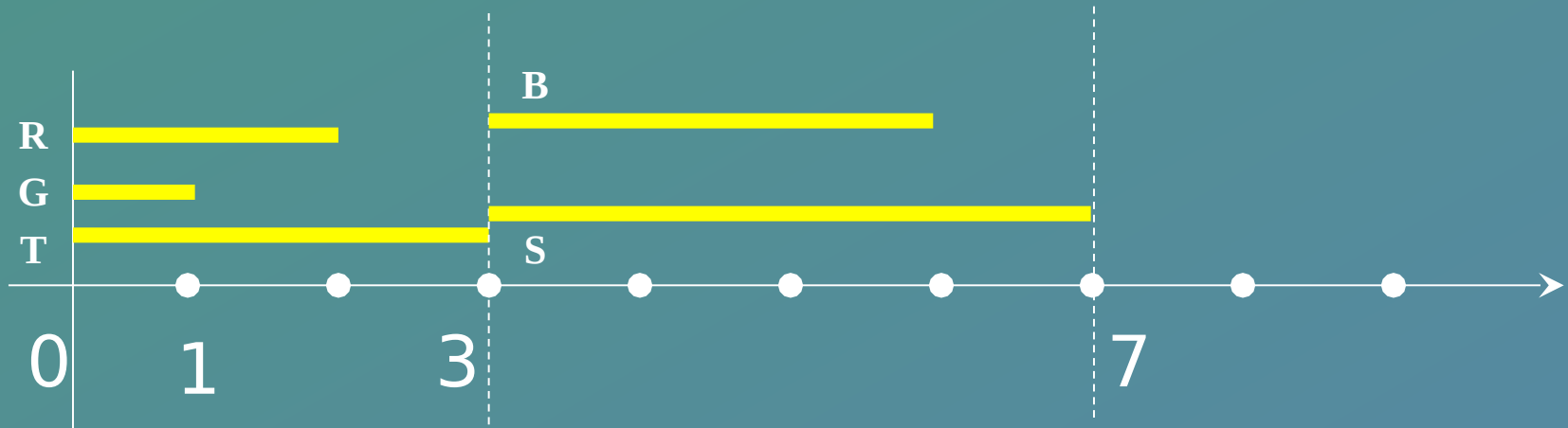
modélisation PERT : une tâche est représentée par un arc



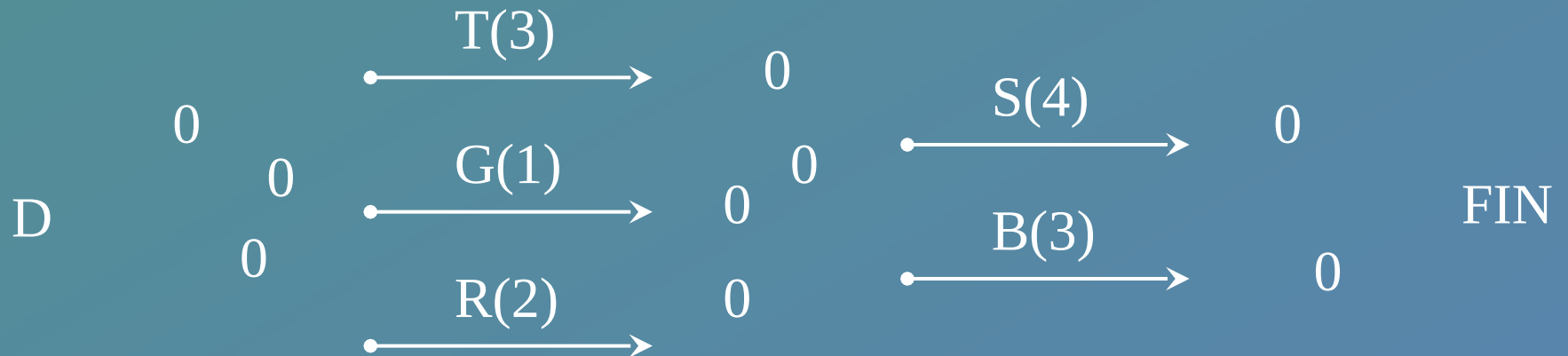


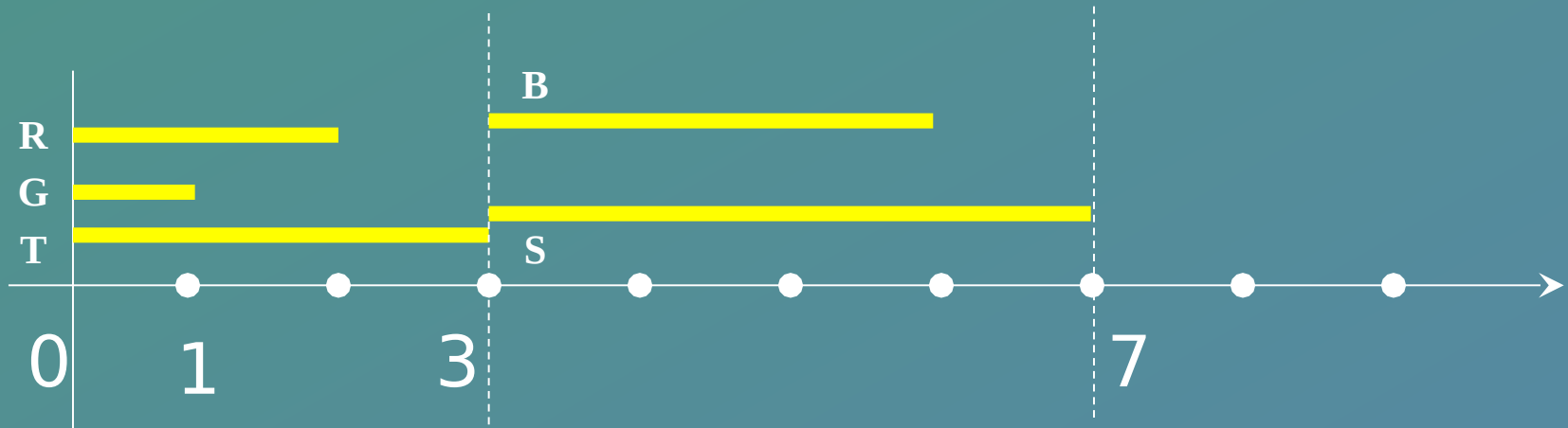
modélisation PERT : une tâche est représentée par un arc



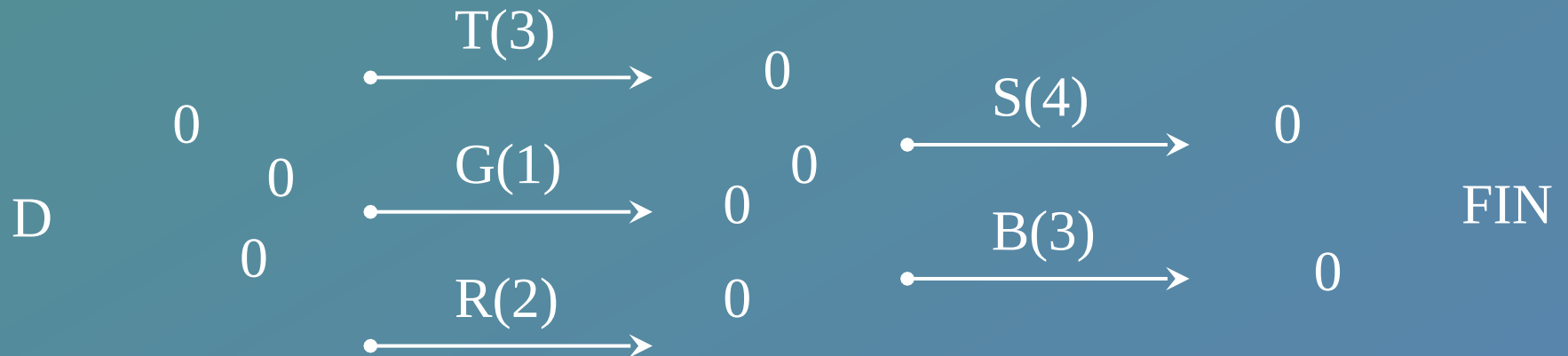


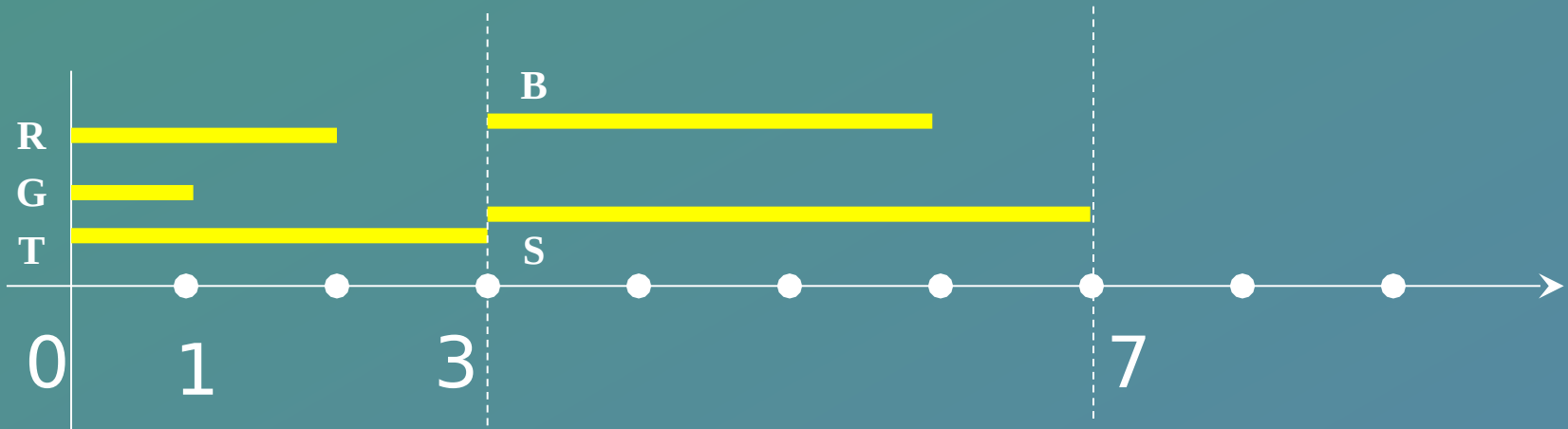
modélisation PERT : une tâche est représentée par un arc



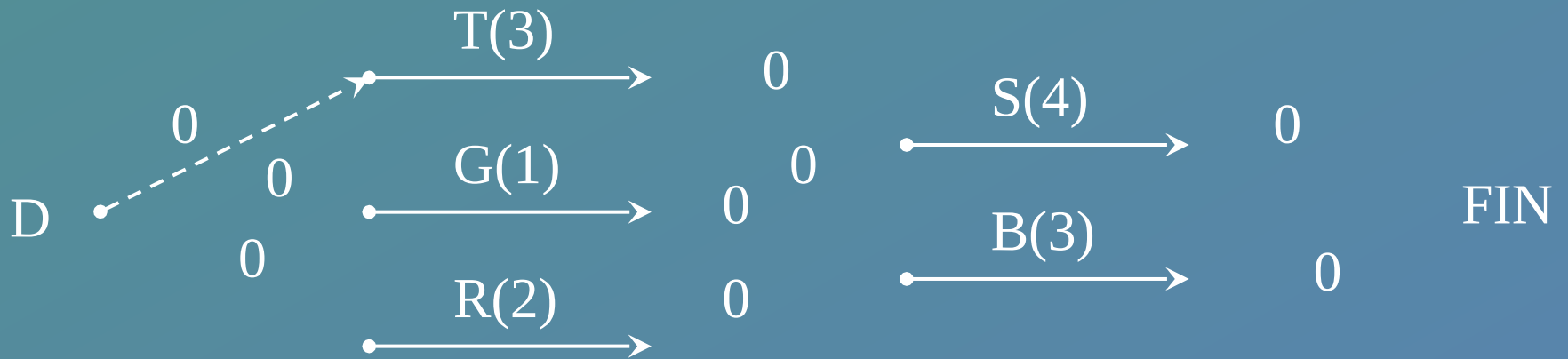


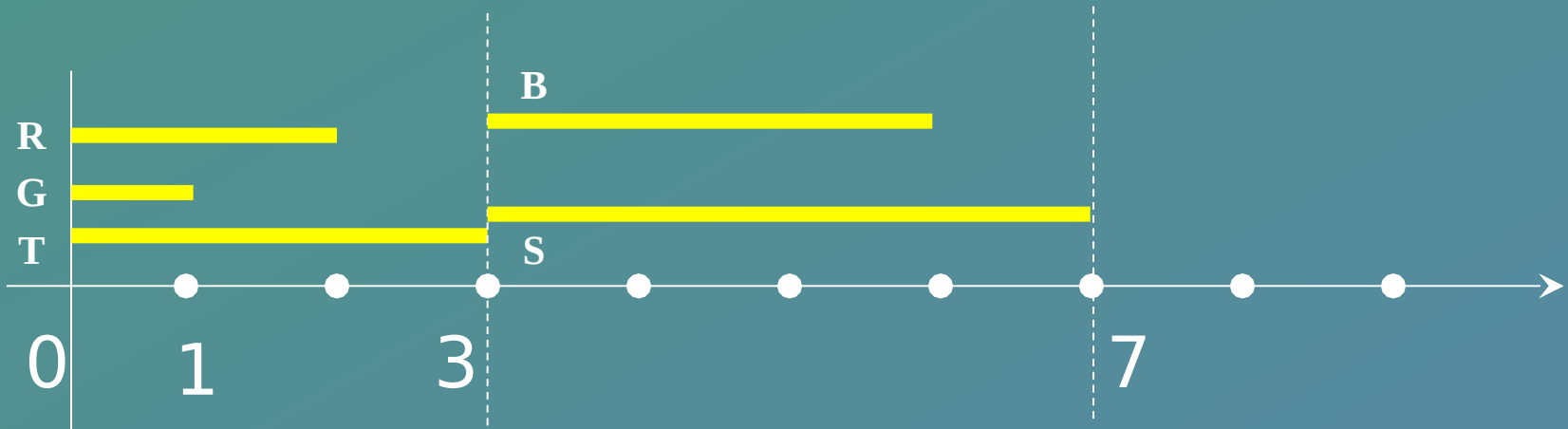
modélisation PERT : une tâche est représentée par un arc



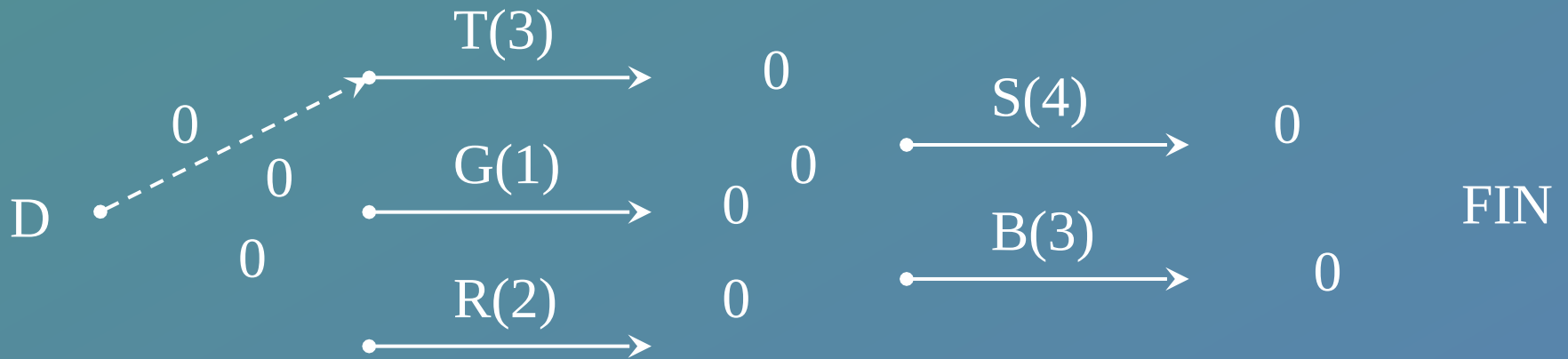


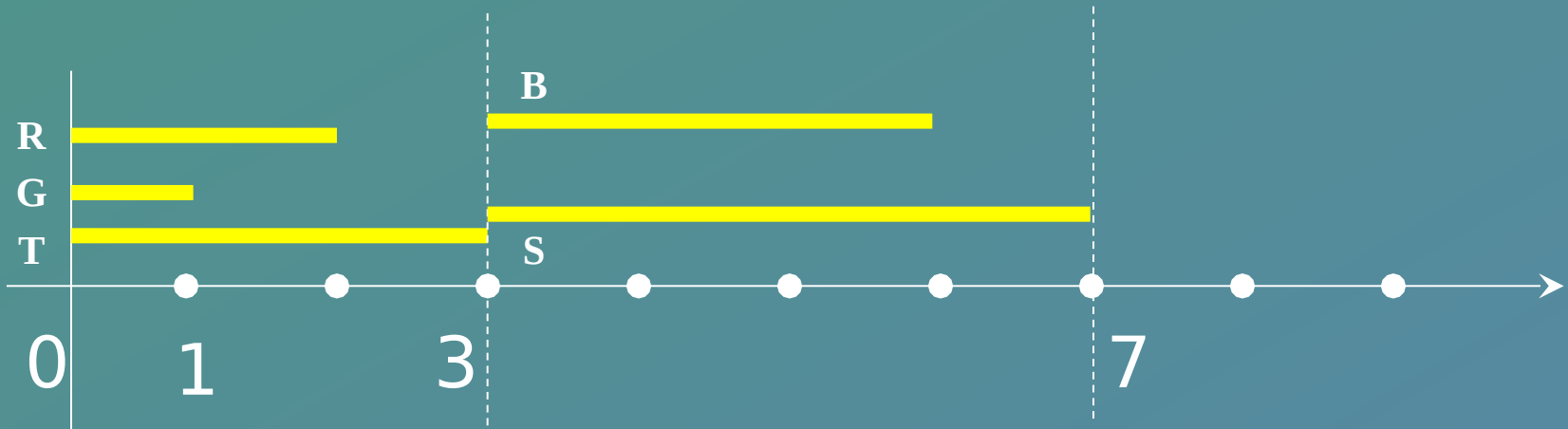
modélisation PERT : une tâche est représentée par un arc



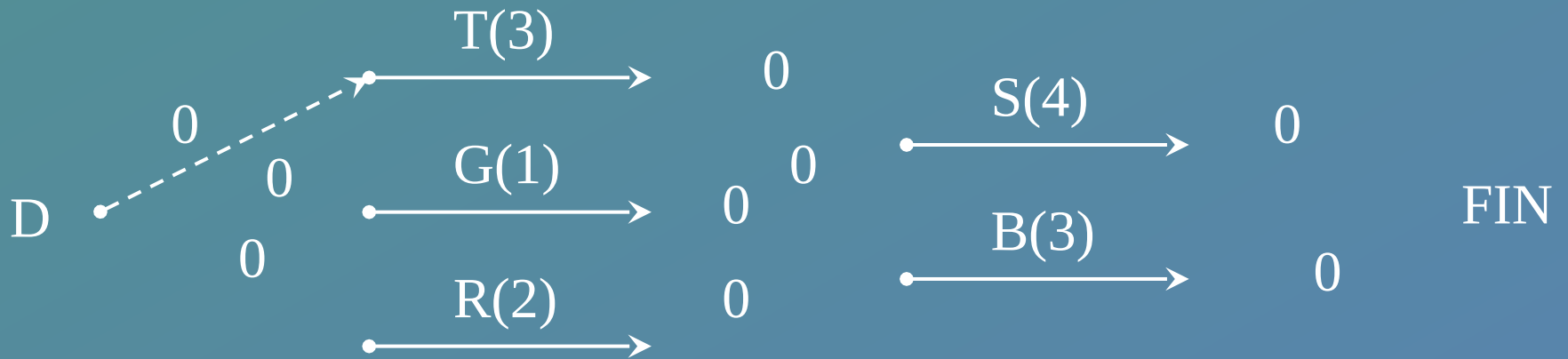


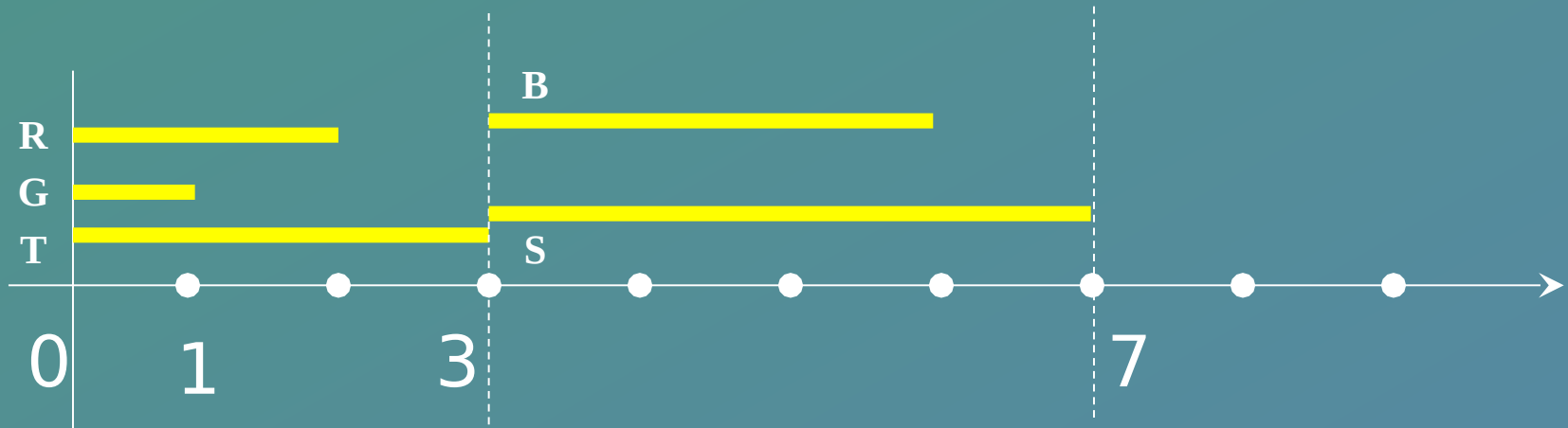
modélisation PERT : une tâche est représentée par un arc



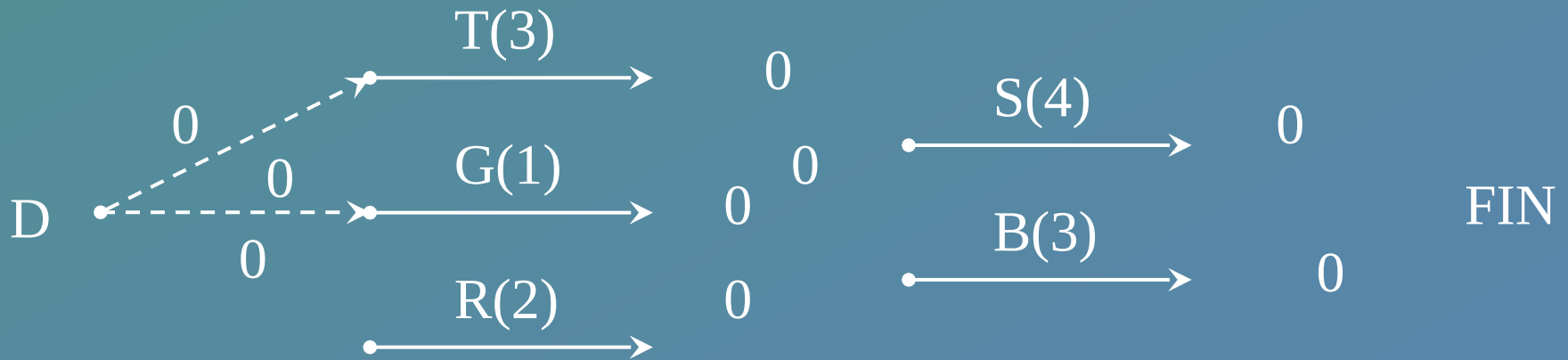


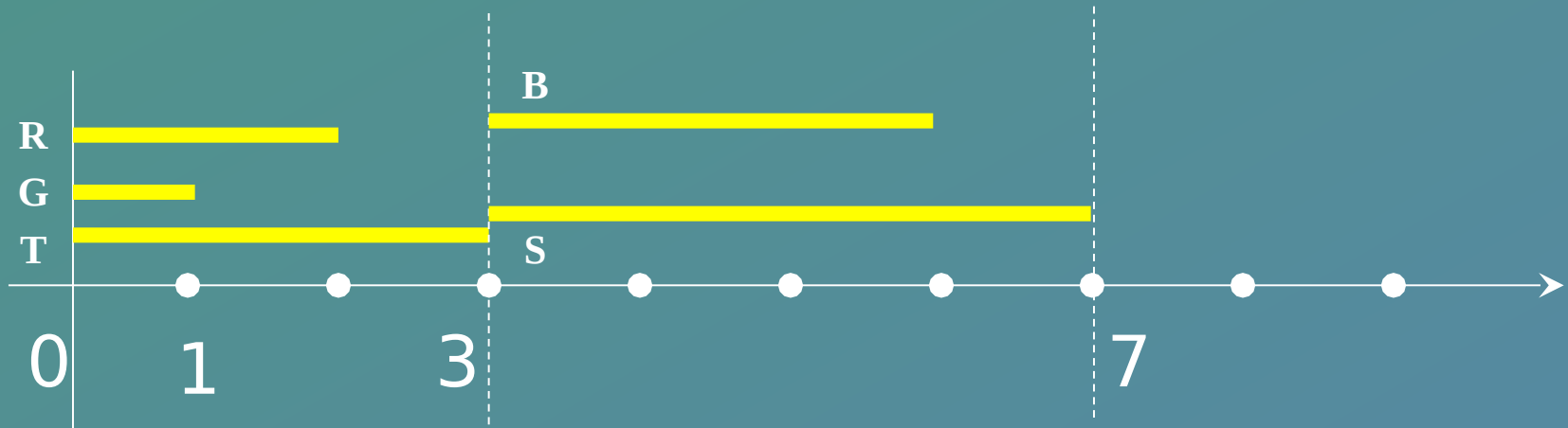
modélisation PERT : une tâche est représentée par un arc



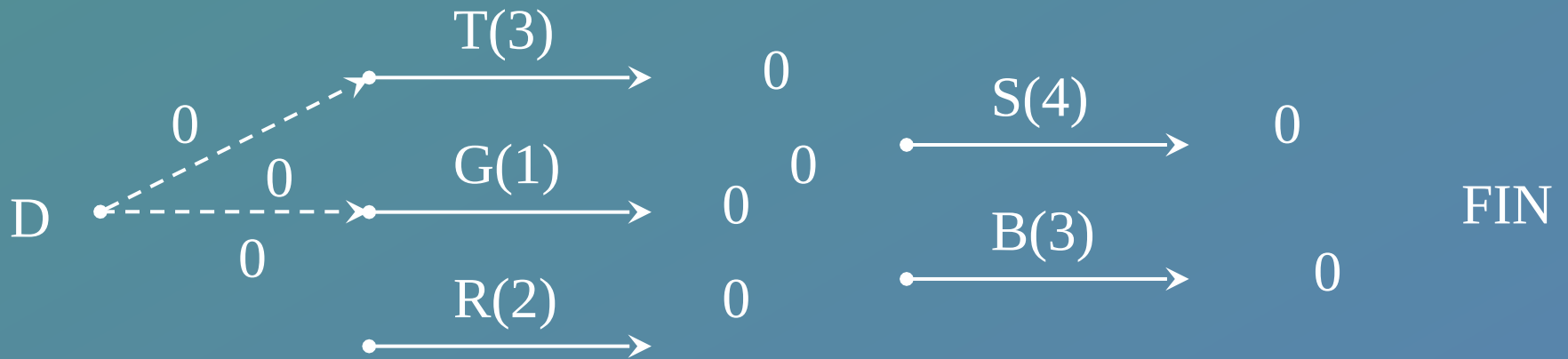


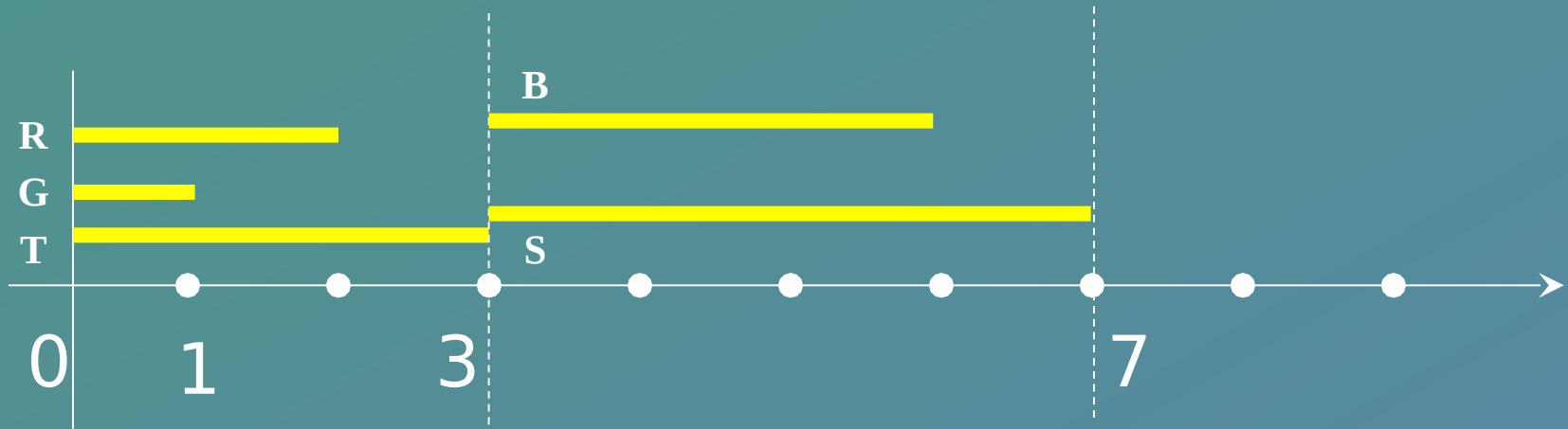
modélisation PERT : une tâche est représentée par un arc



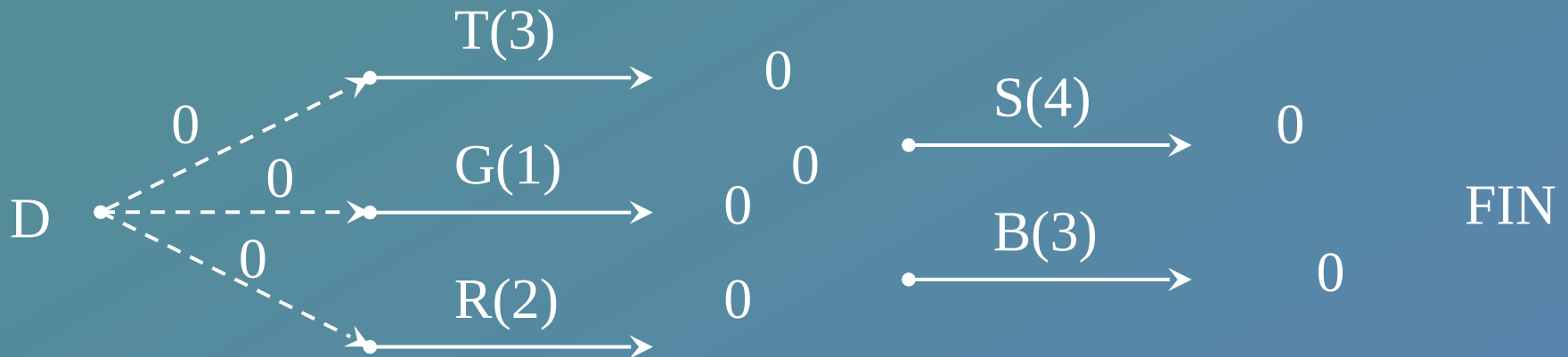


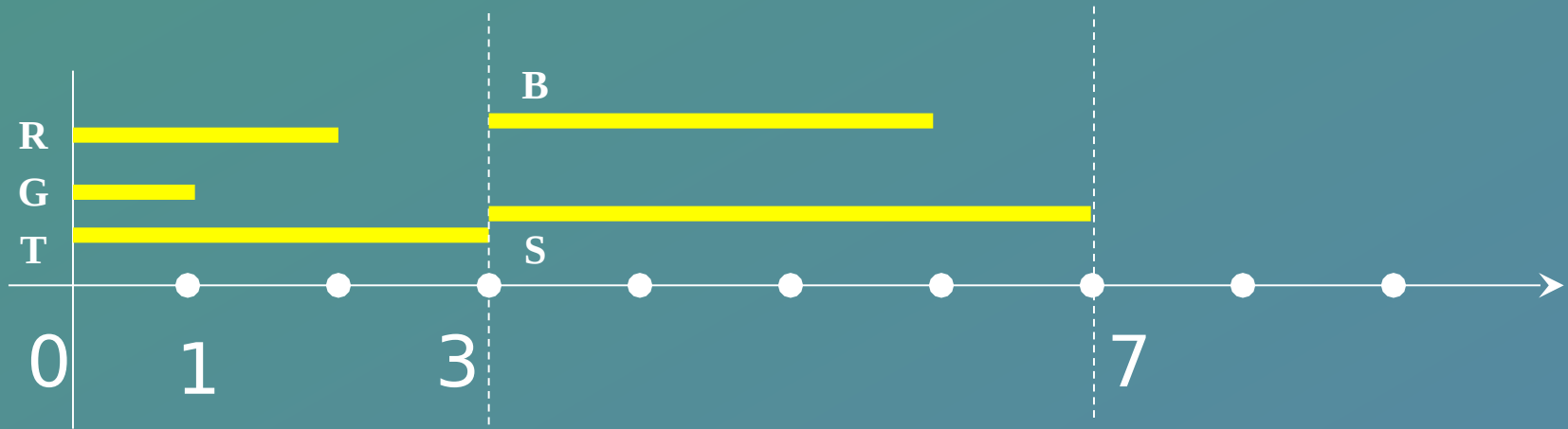
modélisation PERT : une tâche est représentée par un arc



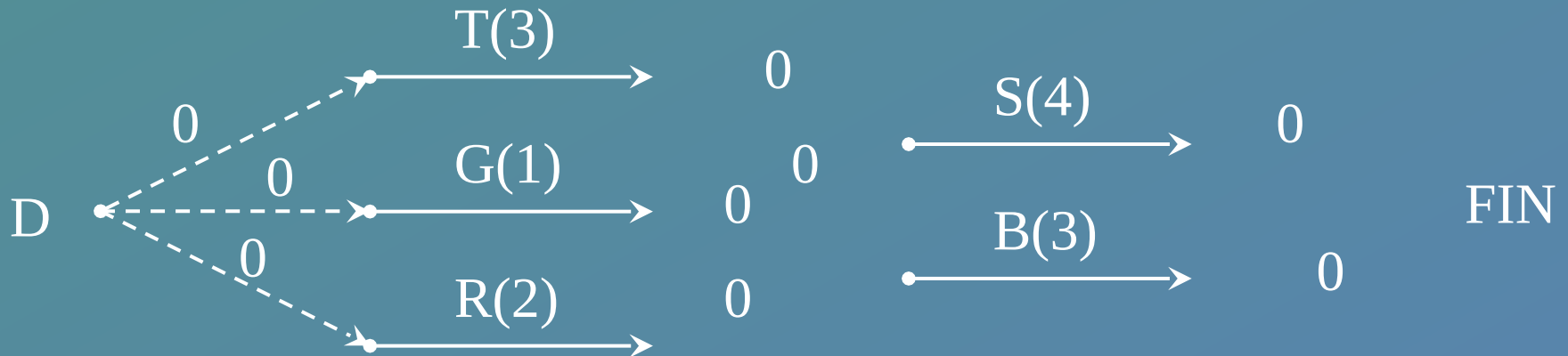


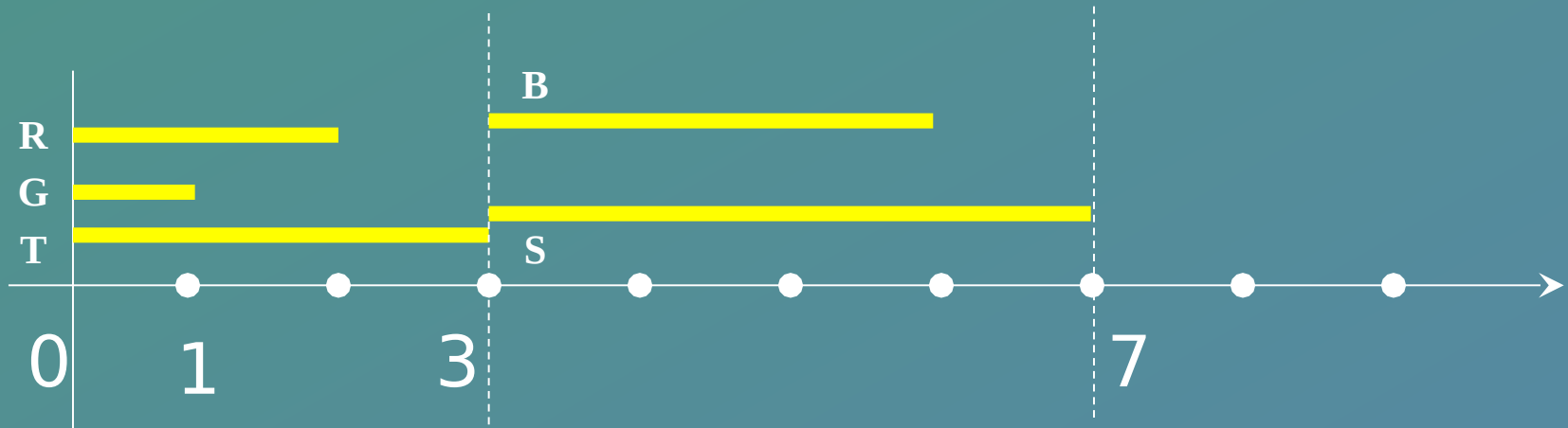
modélisation PERT : une tâche est représentée par un arc



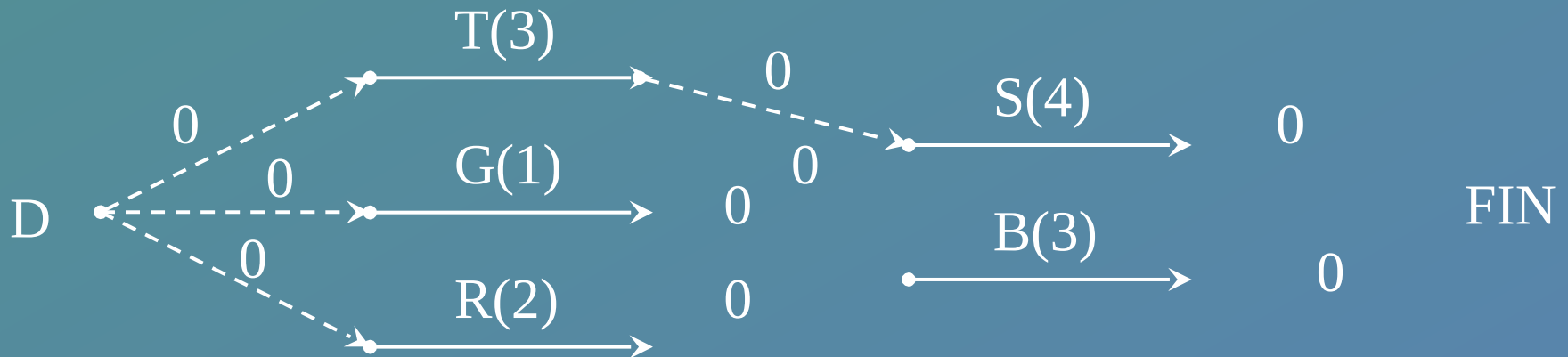


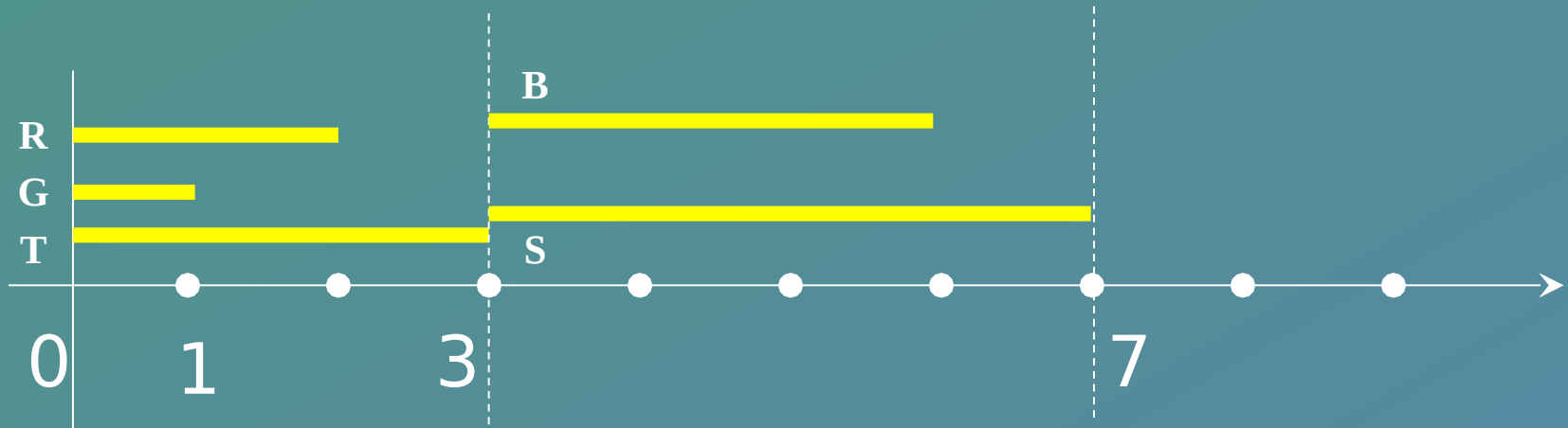
modélisation PERT : une tâche est représentée par un arc



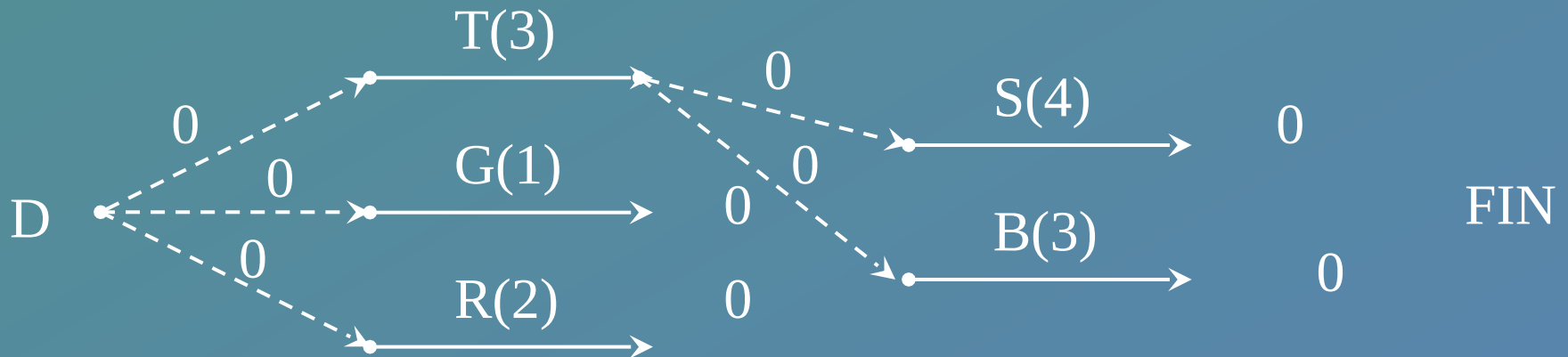


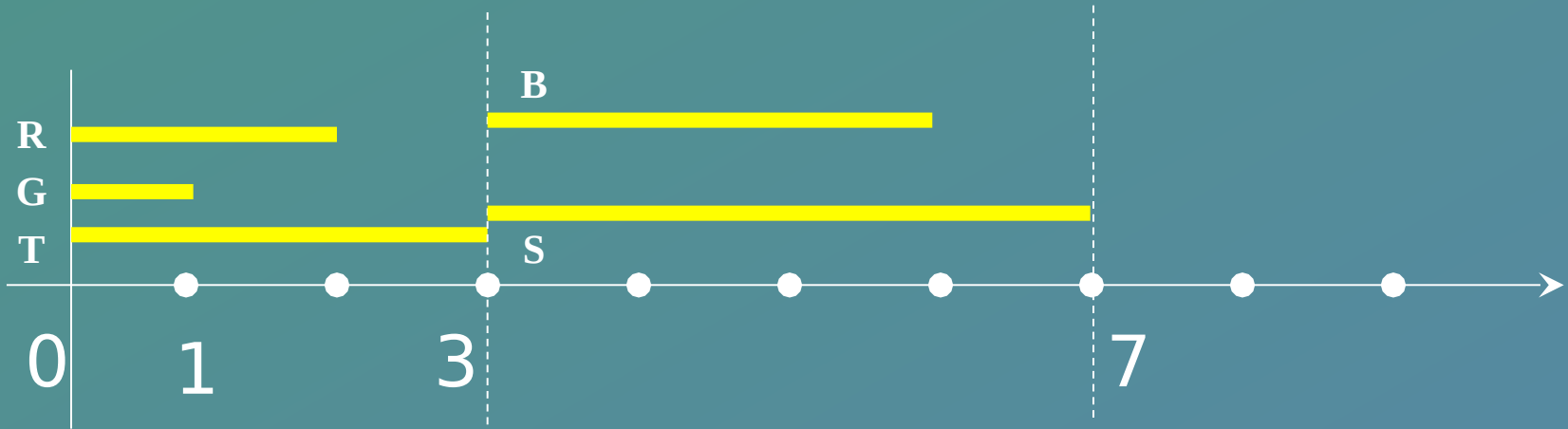
modélisation PERT : une tâche est représentée par un arc



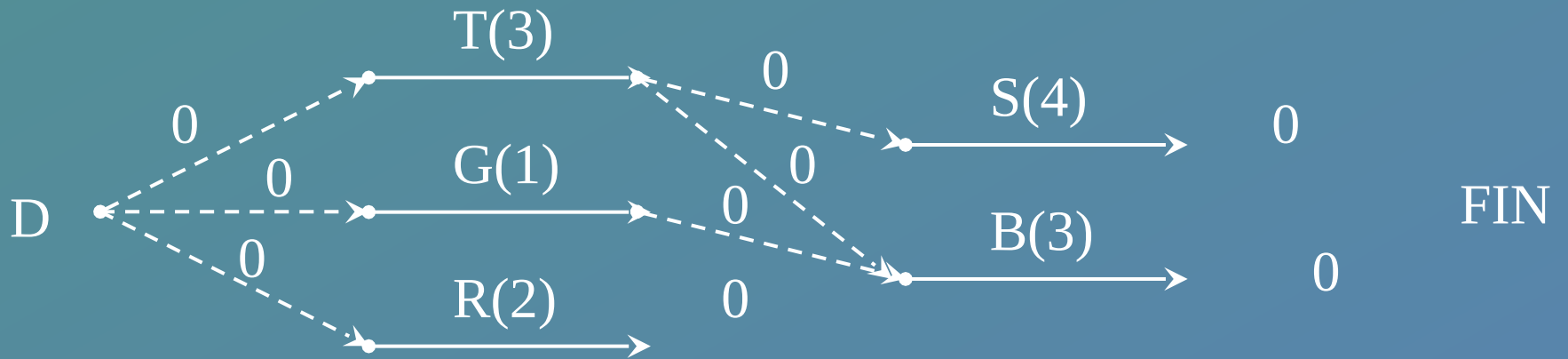


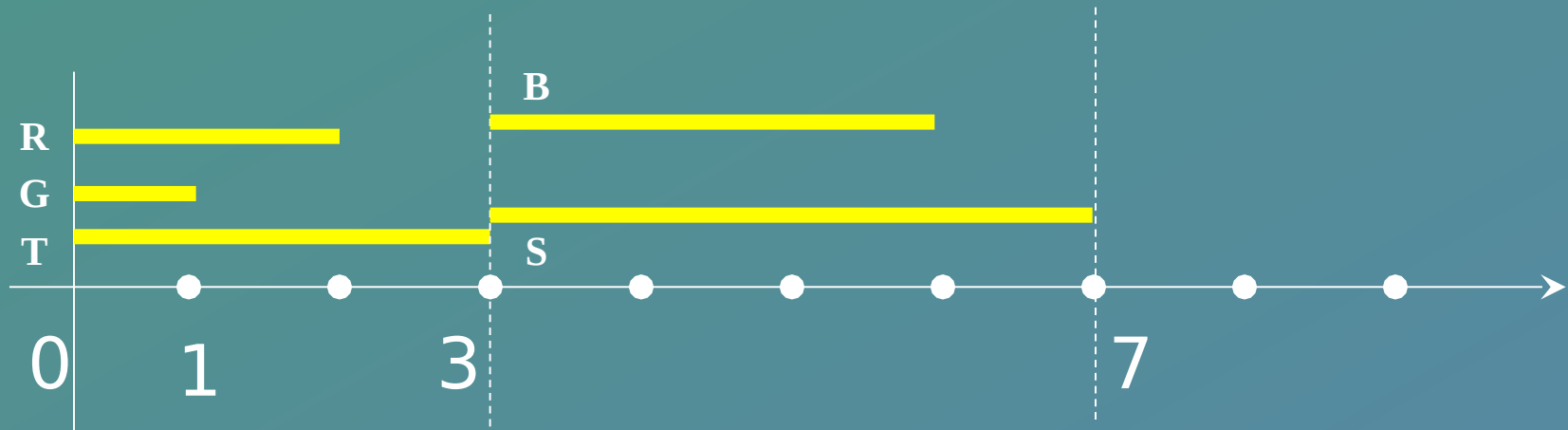
modélisation PERT : une tâche est représentée par un arc



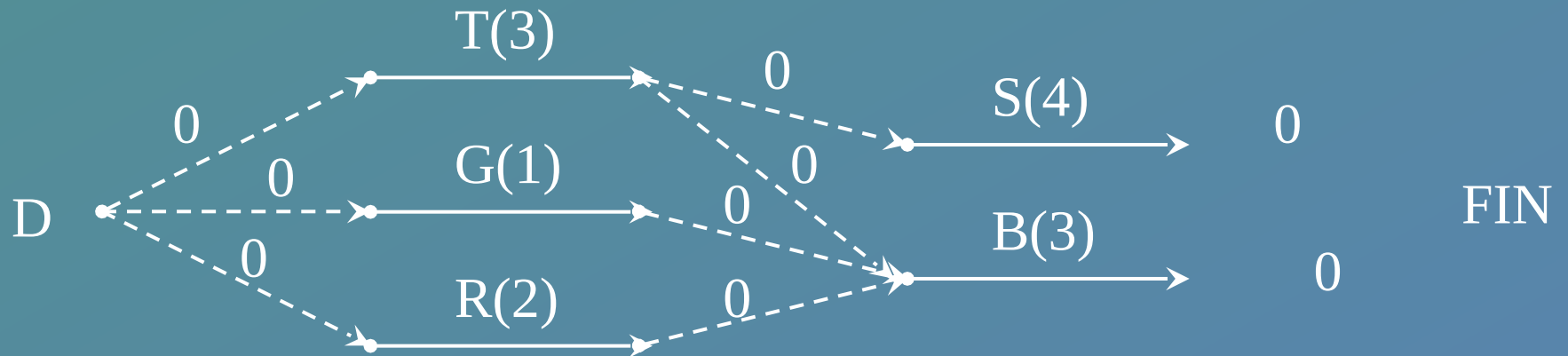


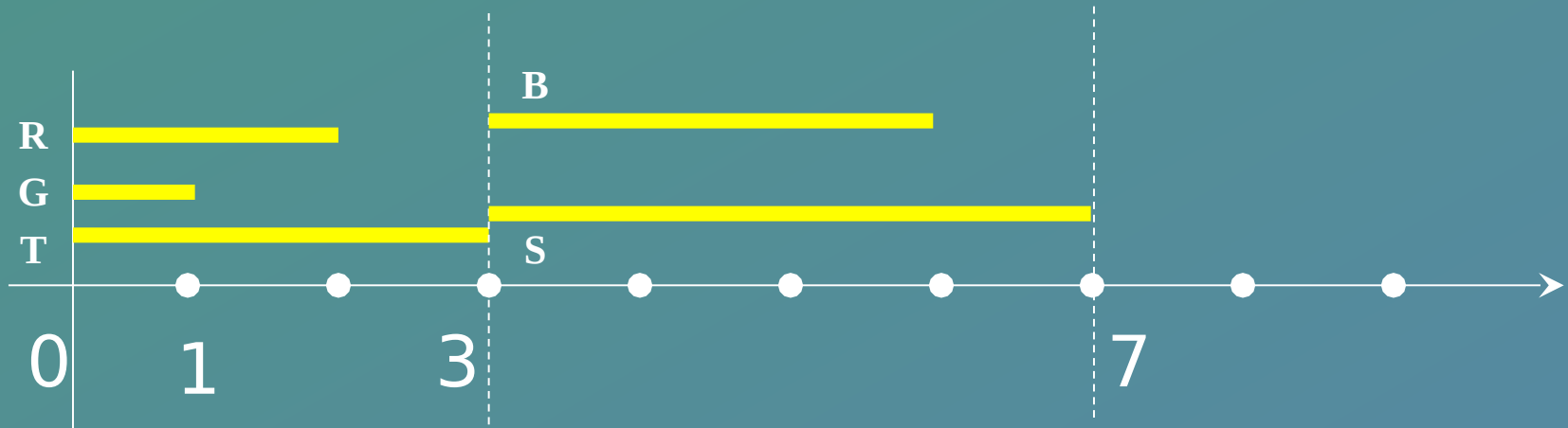
modélisation PERT : une tâche est représentée par un arc



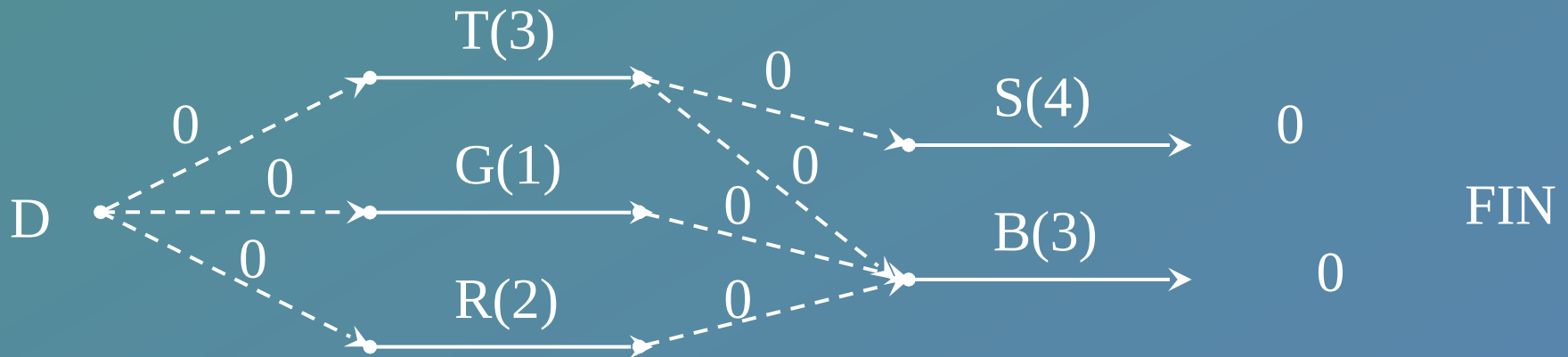


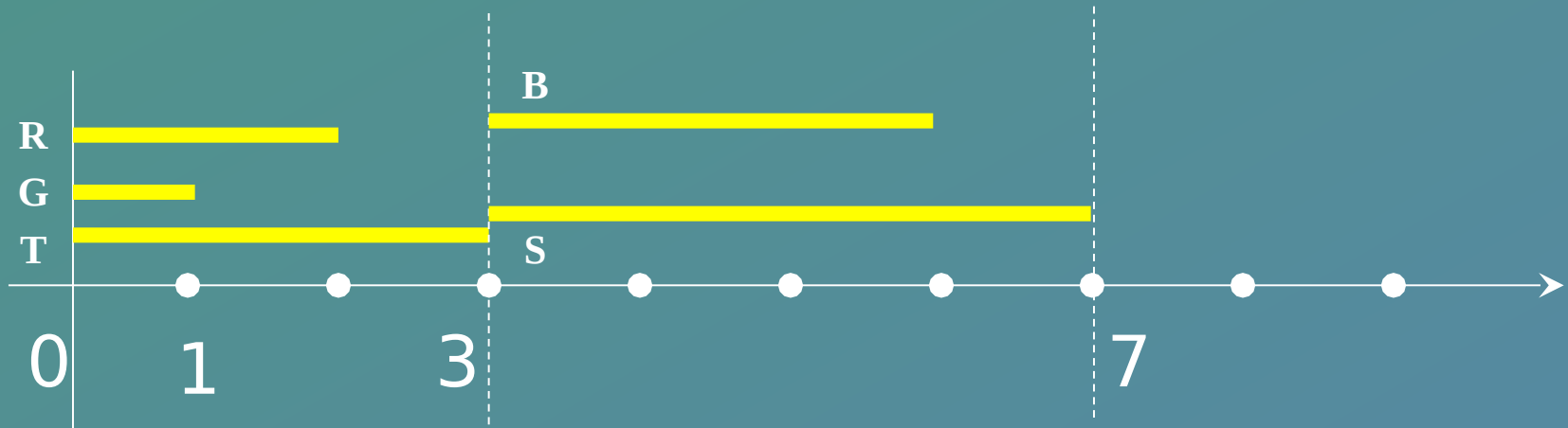
modélisation PERT : une tâche est représentée par un arc



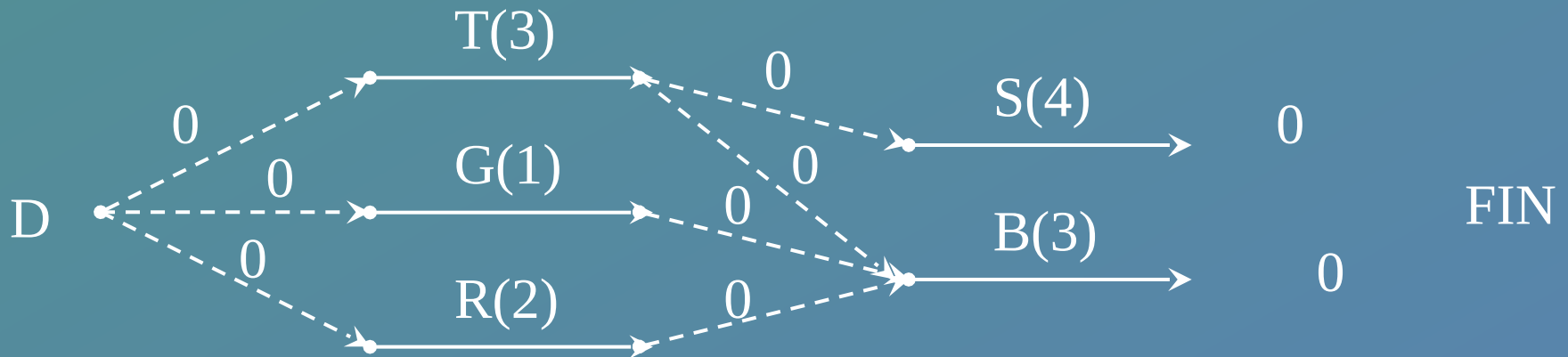


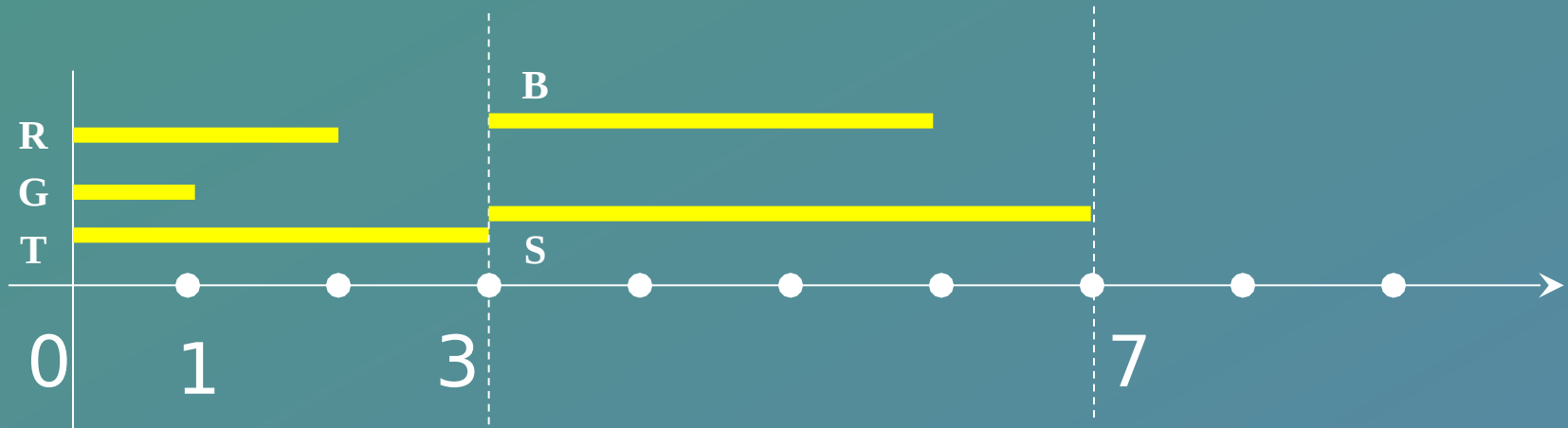
modélisation PERT : une tâche est représentée par un arc



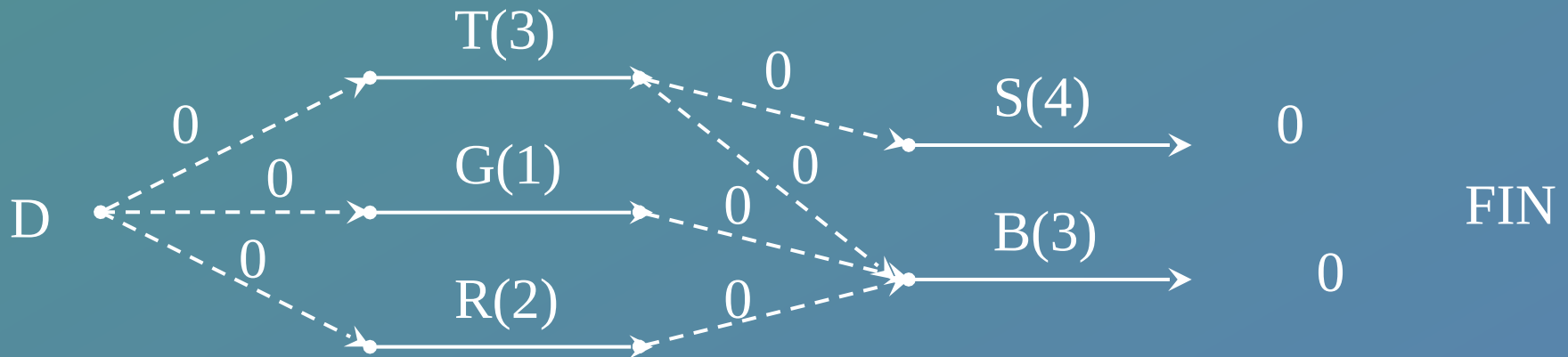


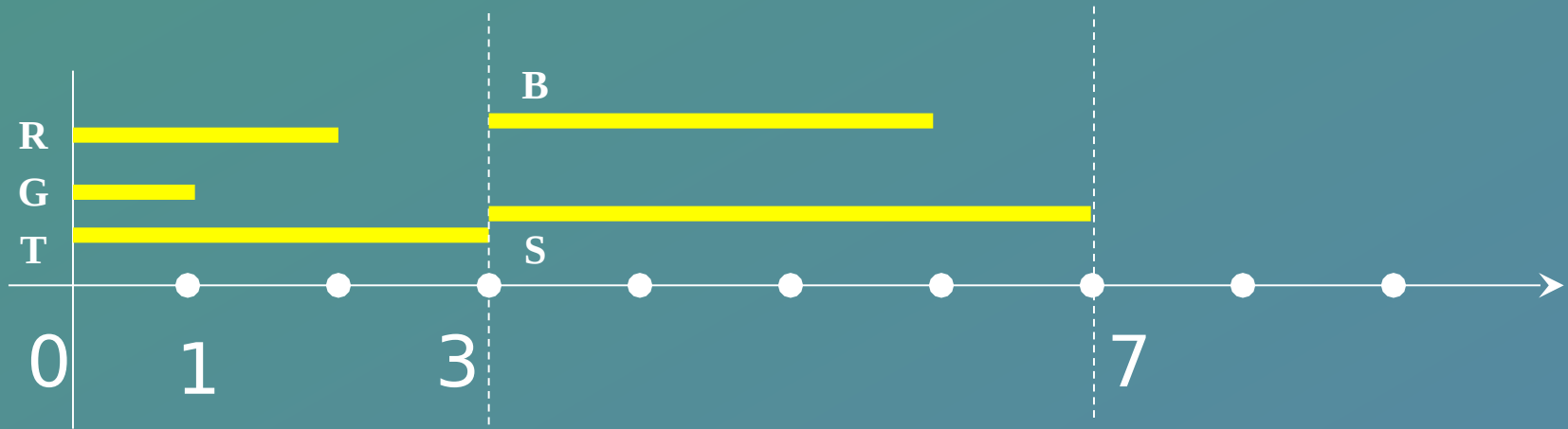
modélisation PERT : une tâche est représentée par un arc



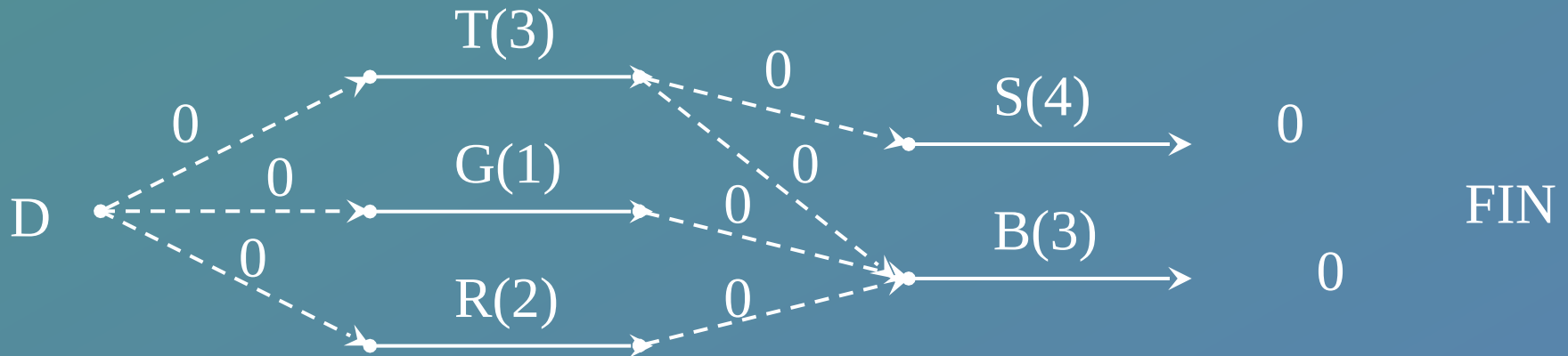


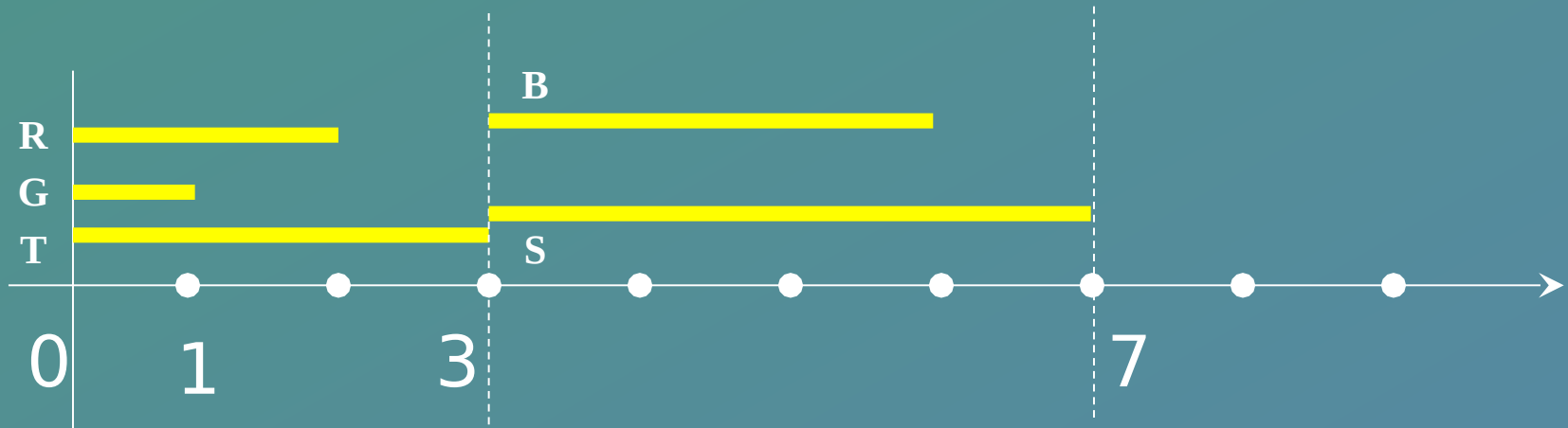
modélisation PERT : une tâche est représentée par un arc



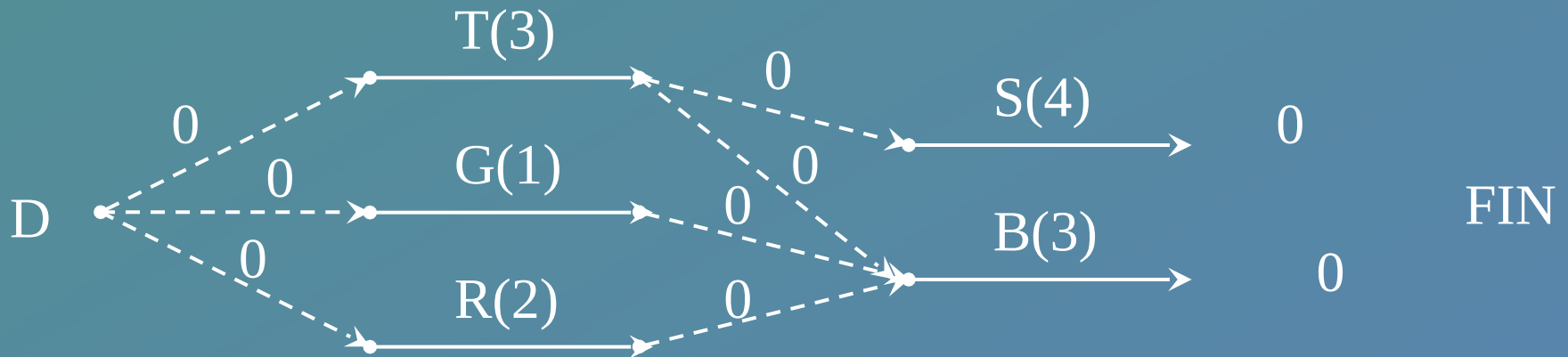


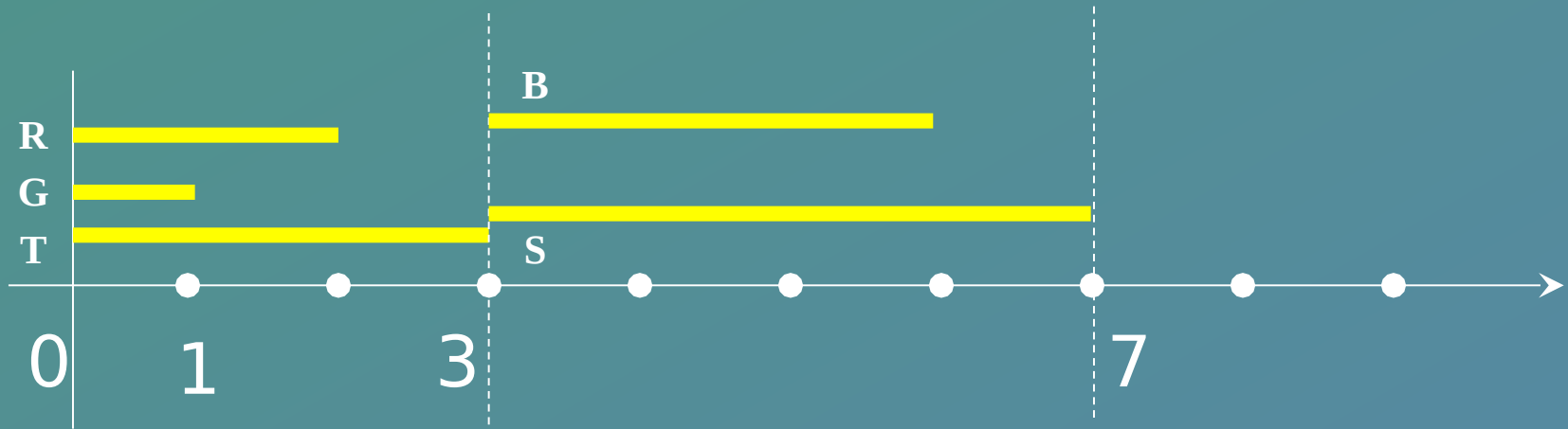
modélisation PERT : une tâche est représentée par un arc



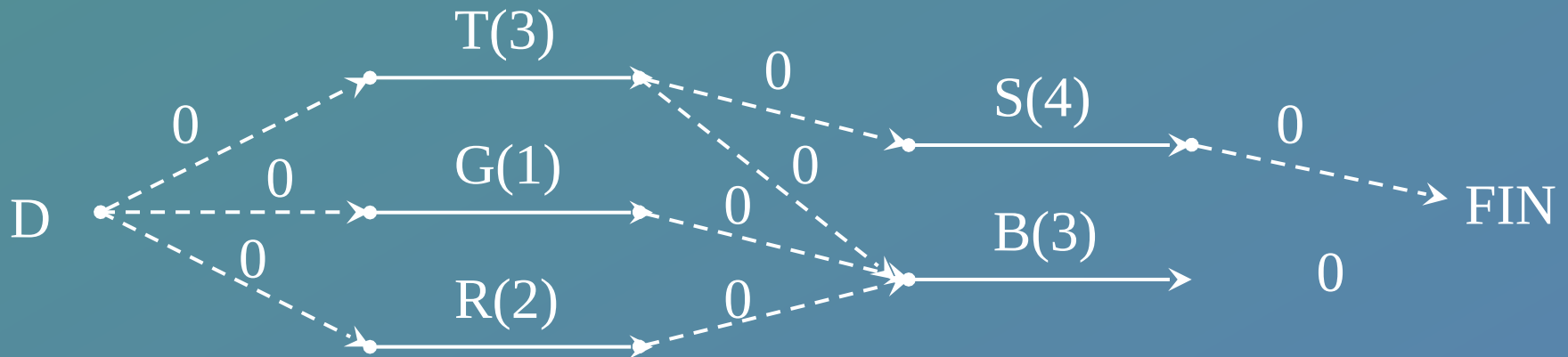


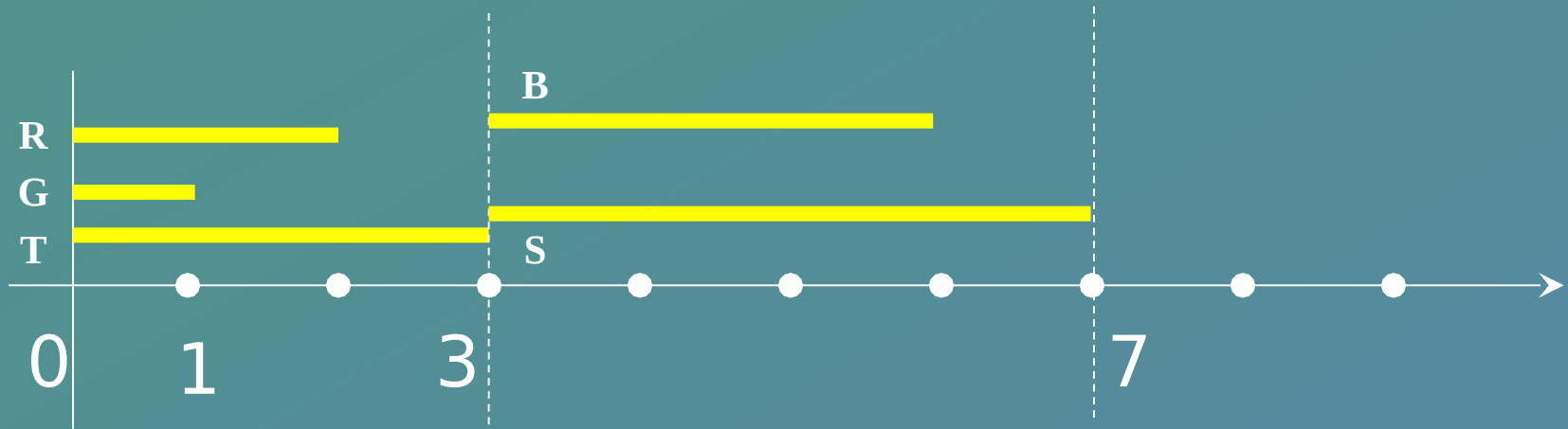
modélisation PERT : une tâche est représentée par un arc



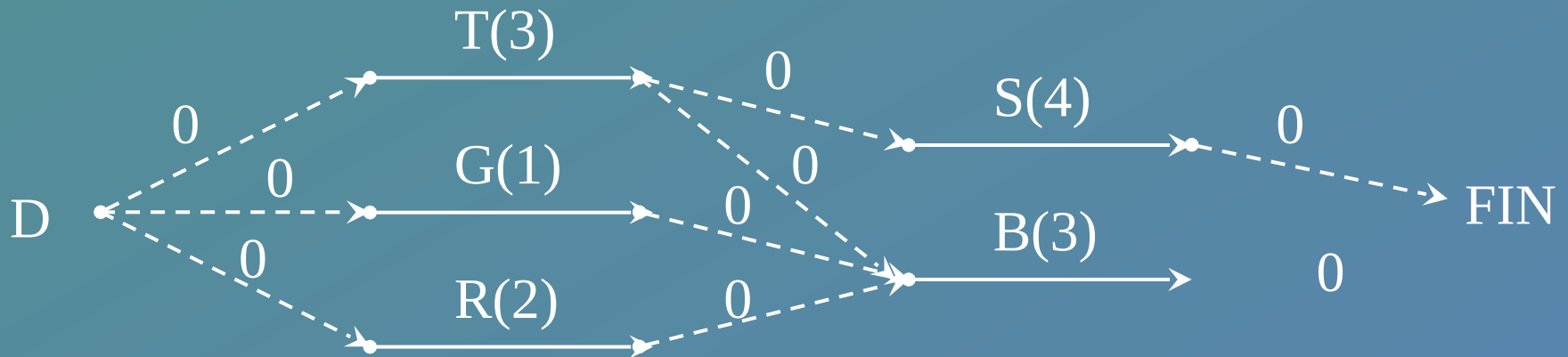


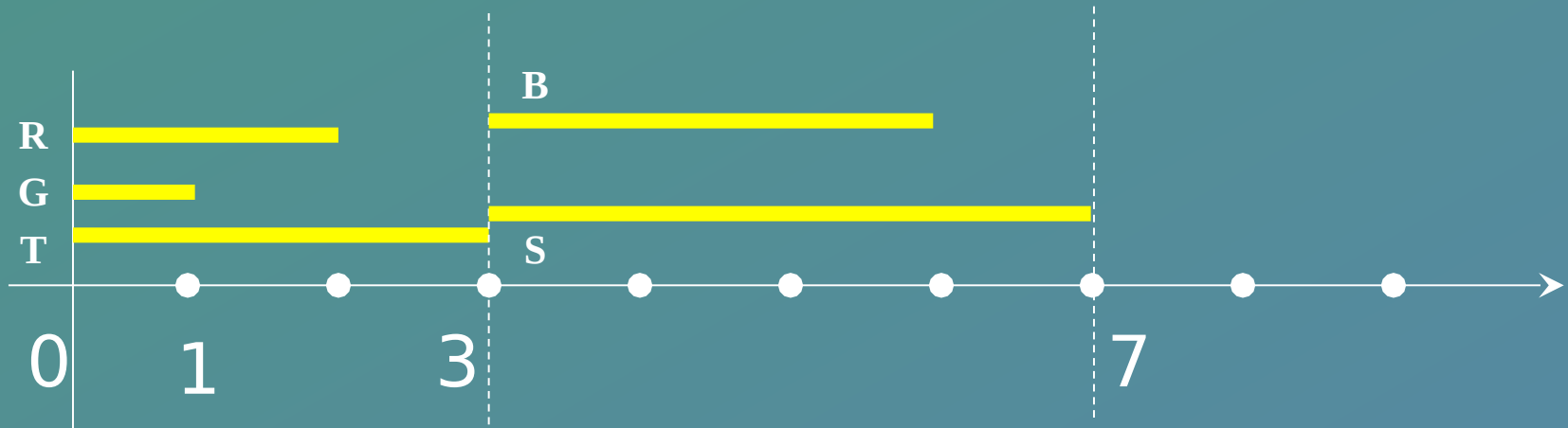
modélisation PERT : une tâche est représentée par un arc



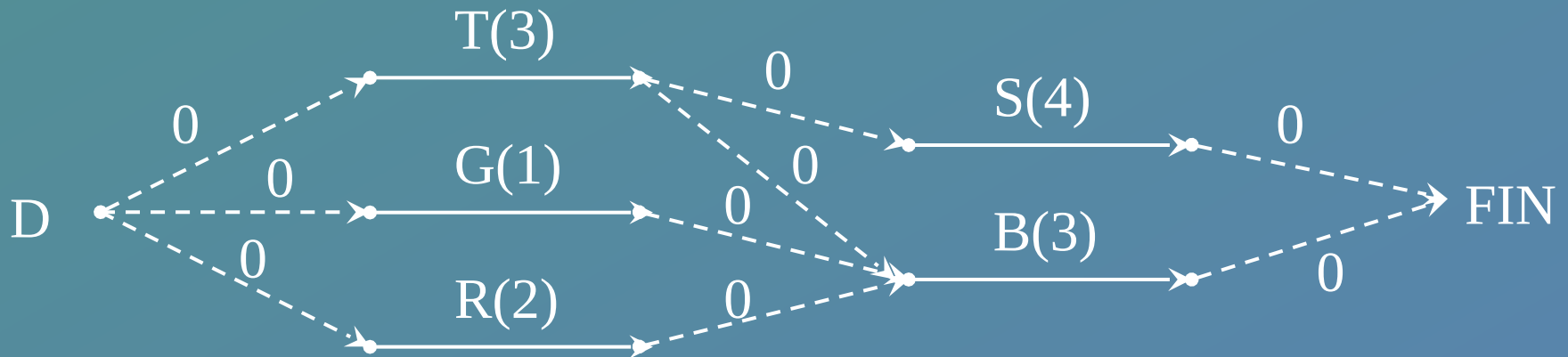


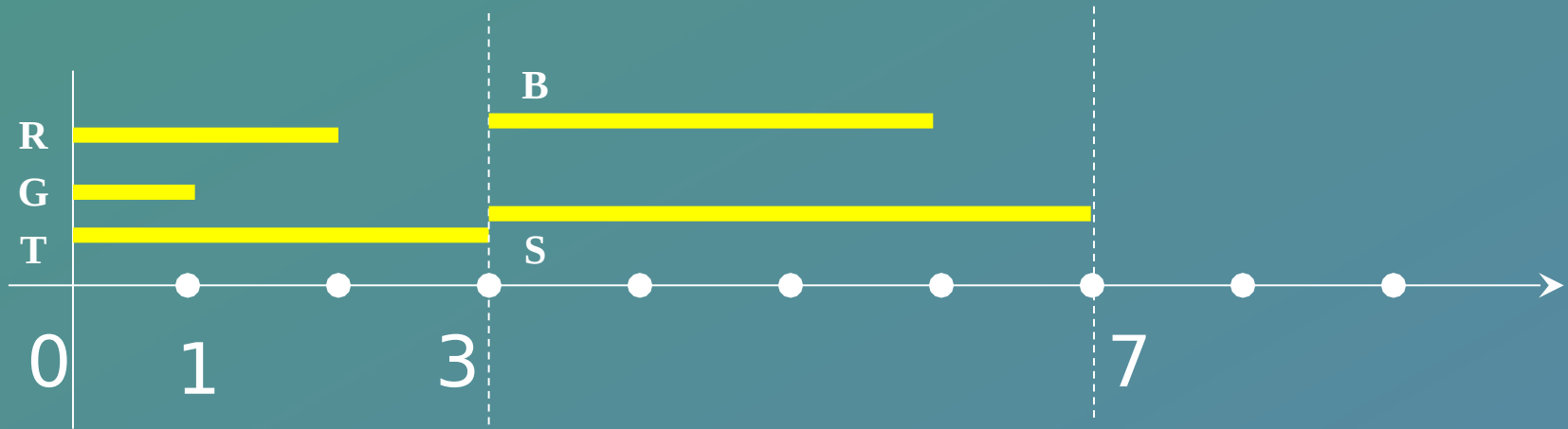
modélisation PERT : une tâche est représentée par un arc



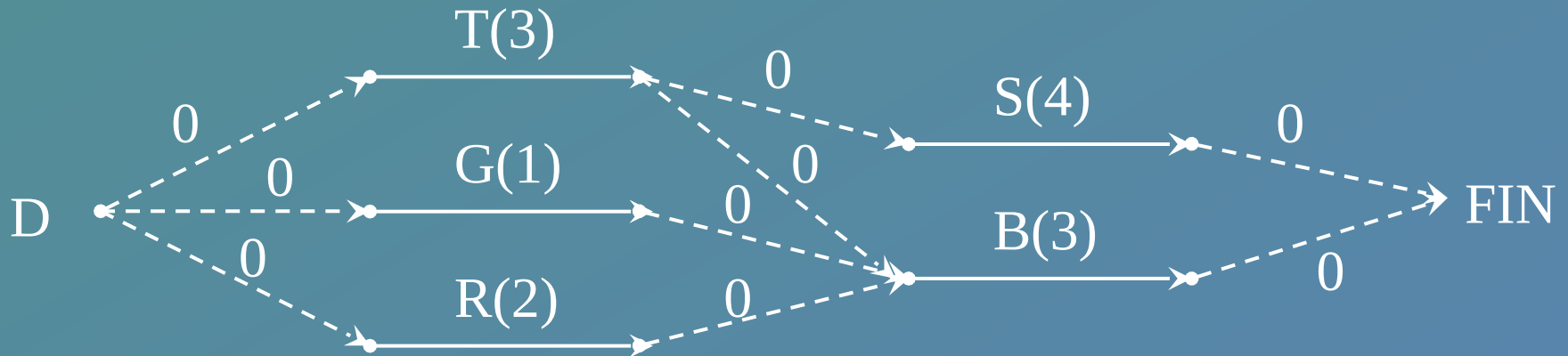


modélisation PERT : une tâche est représentée par un arc





modélisation PERT : une tâche est représentée par un arc



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



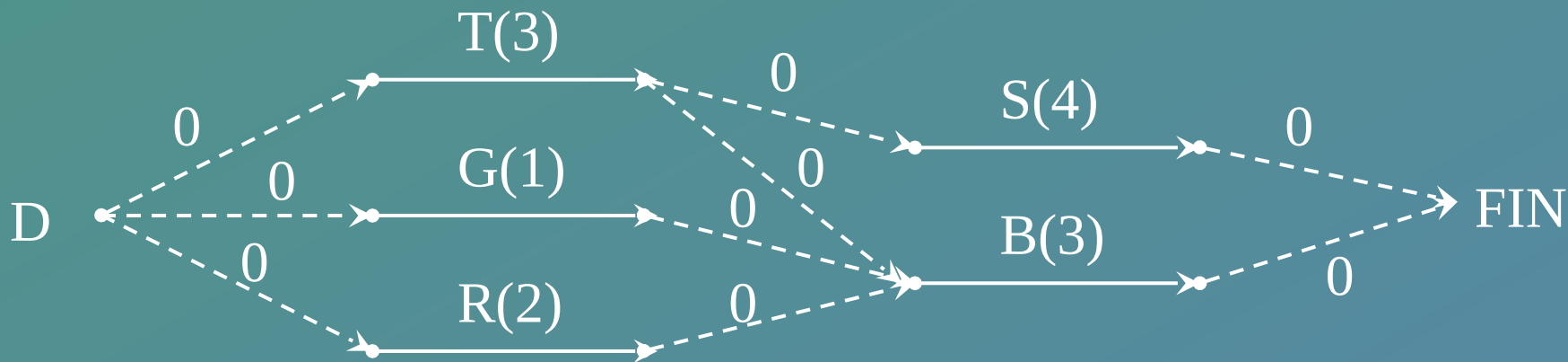
Ce graphe, beaucoup trop compliqué, est sans intérêt pratique et il faut obligatoirement appliquer une procédure de simplification :

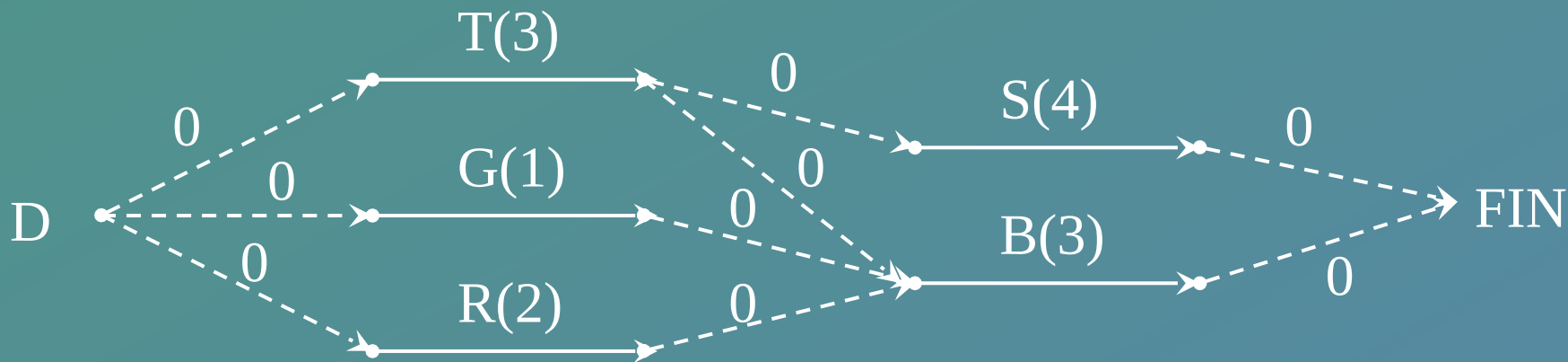
On peut identifier (contracter) deux extrémités d'une tâche fictive et enlever la tâche fictive.



Il faut prendre garde que chaque opération de réduction préserve toutes les contraintes d'antériorité et n'en ajoute pas de nouvelles.

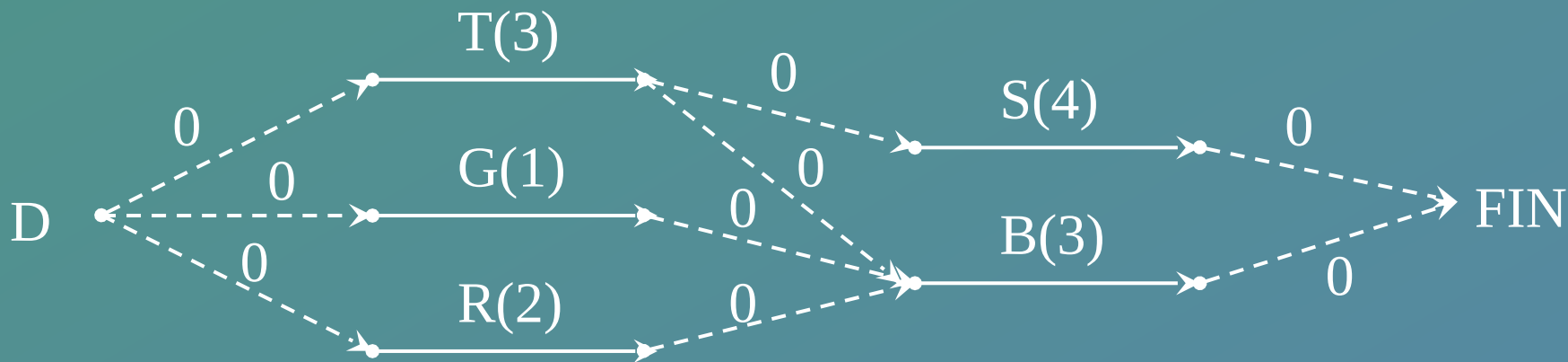






D

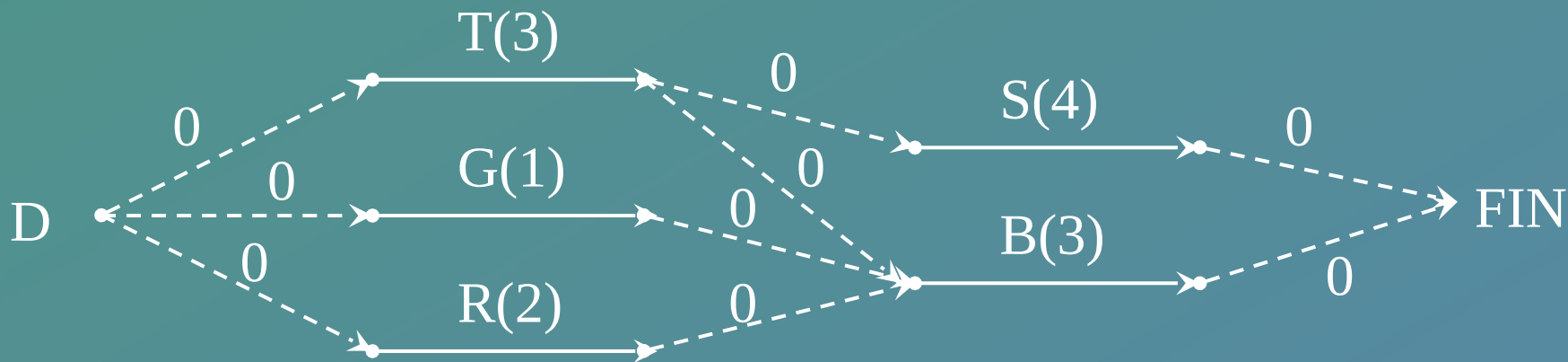


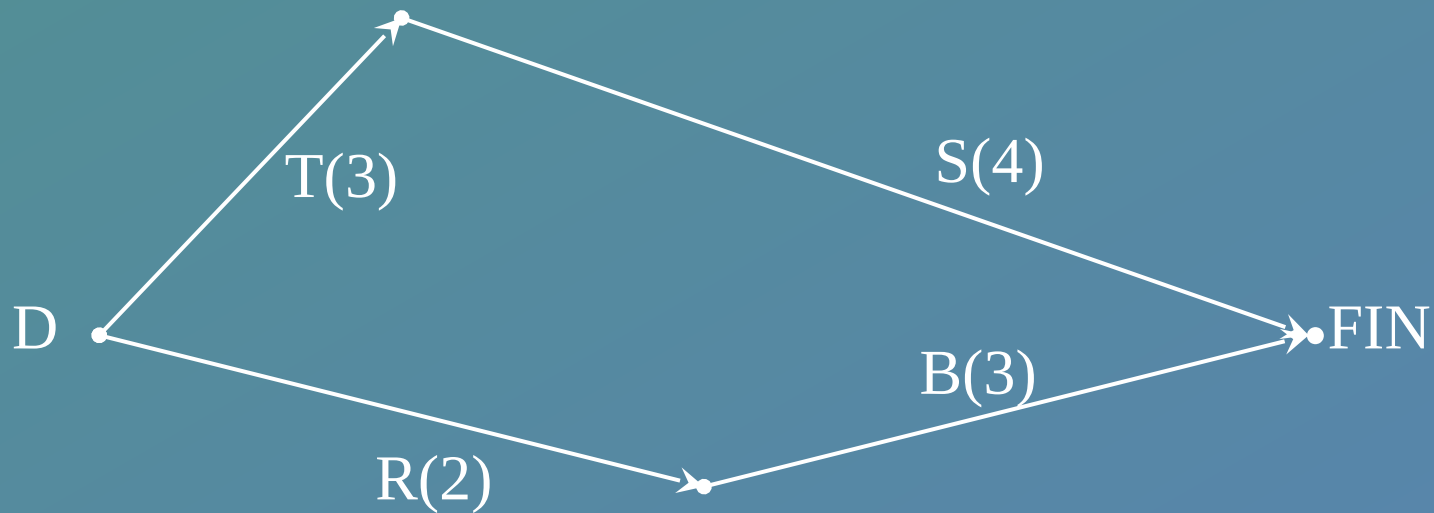
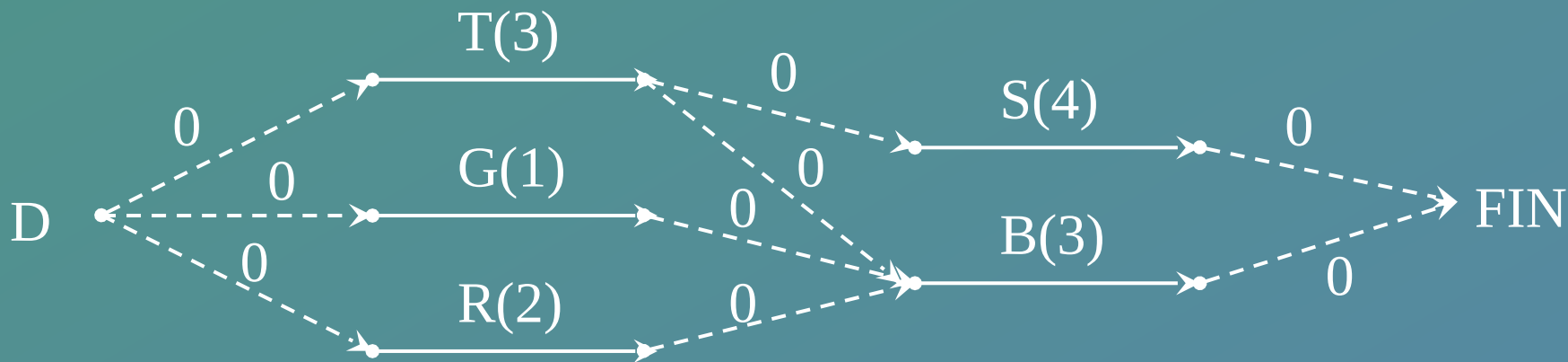


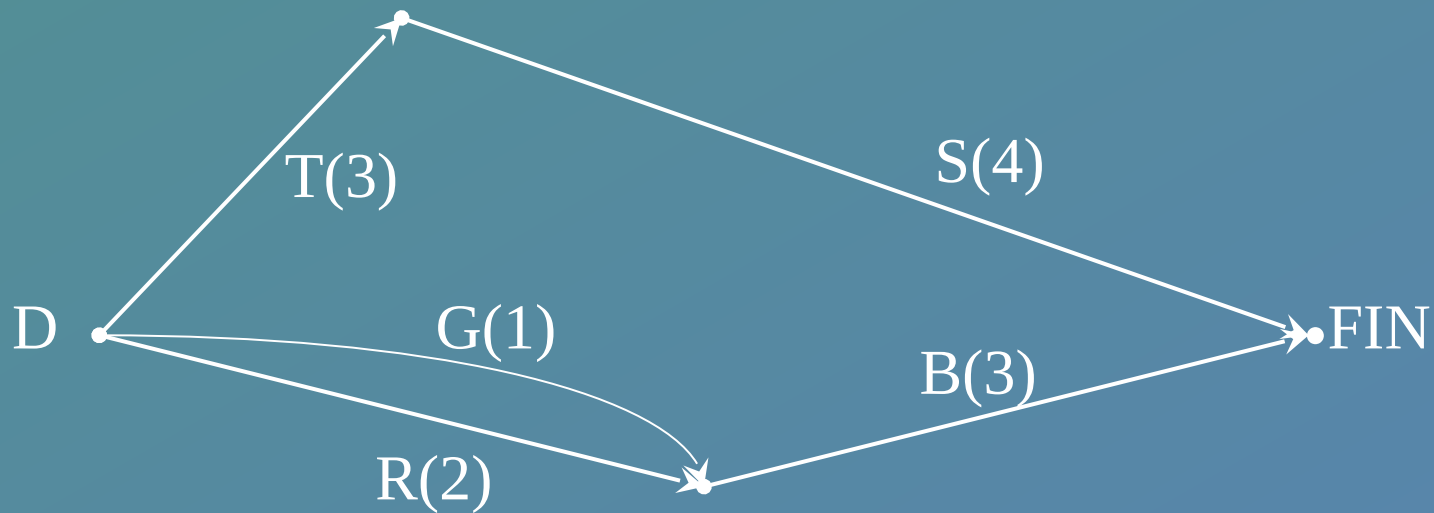
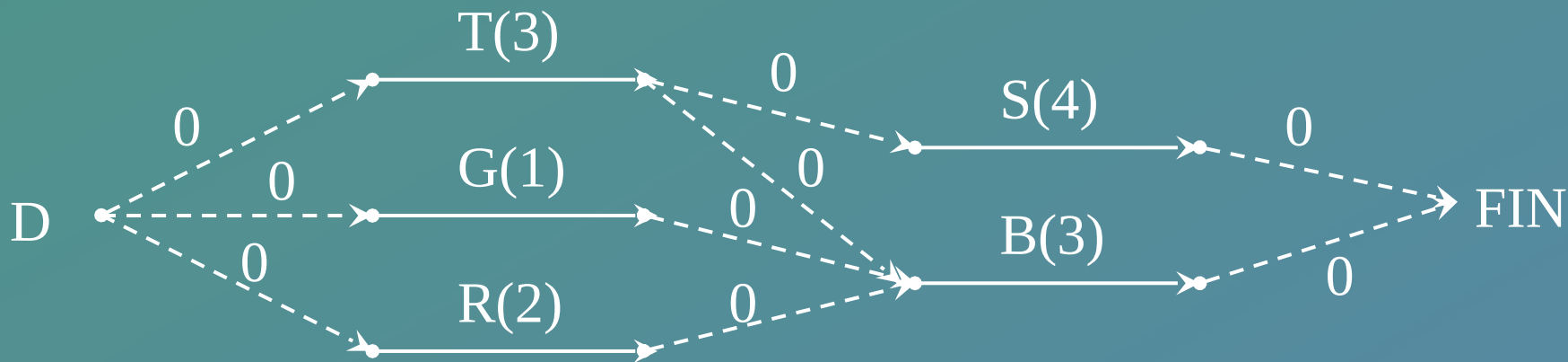
D

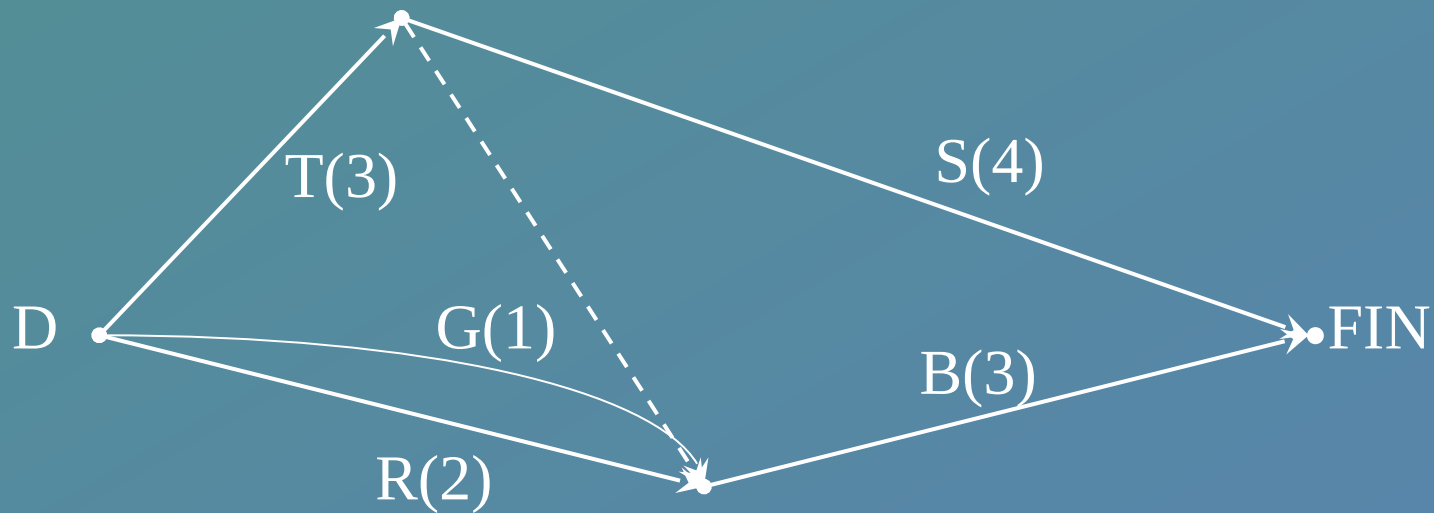
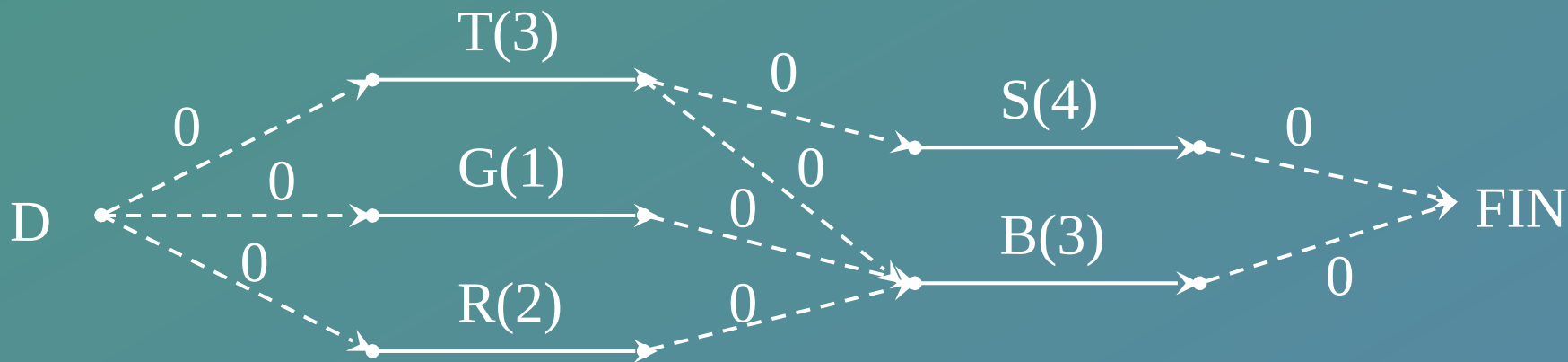
•FIN

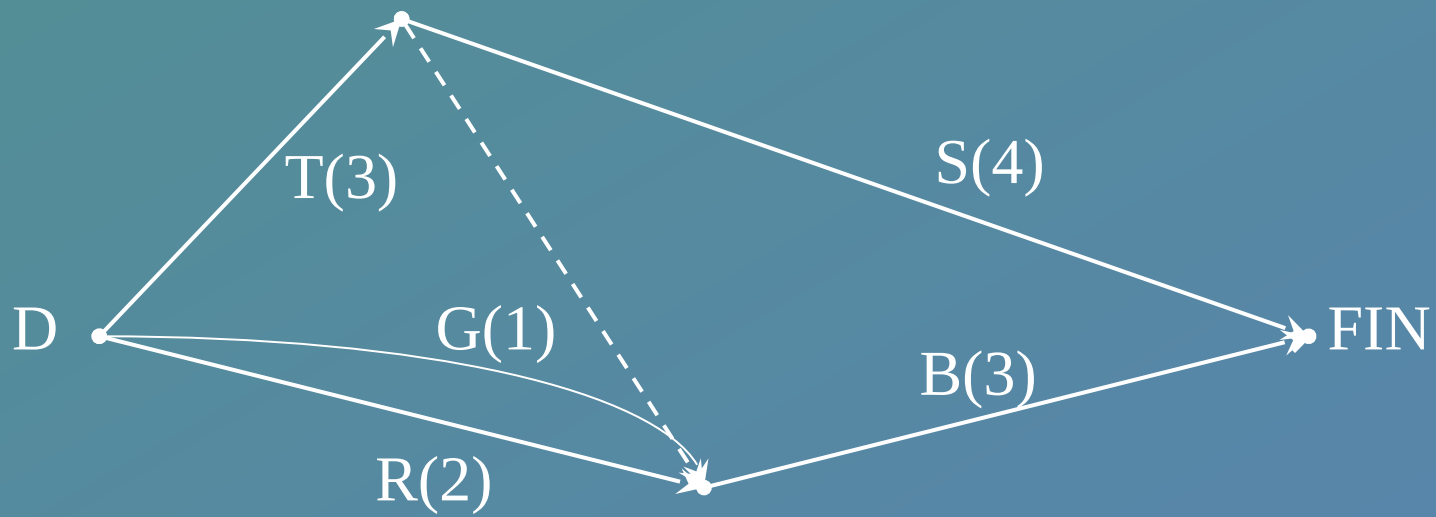


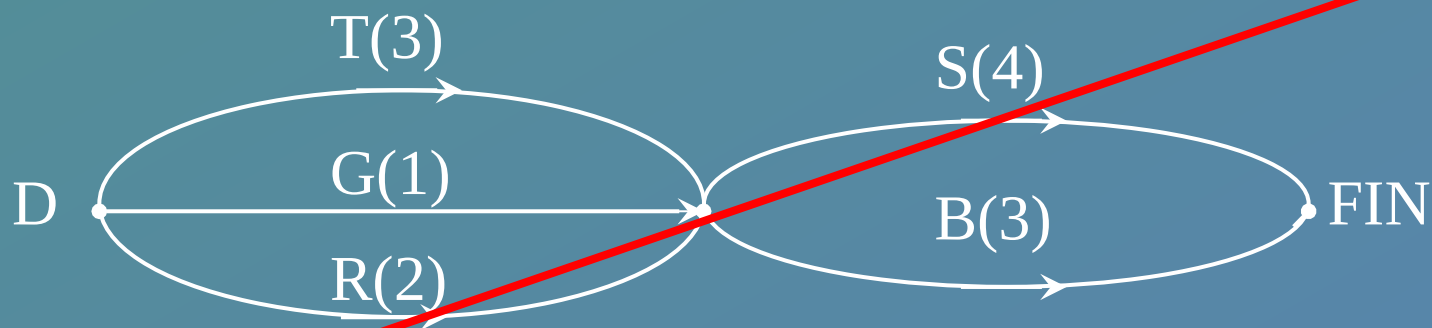
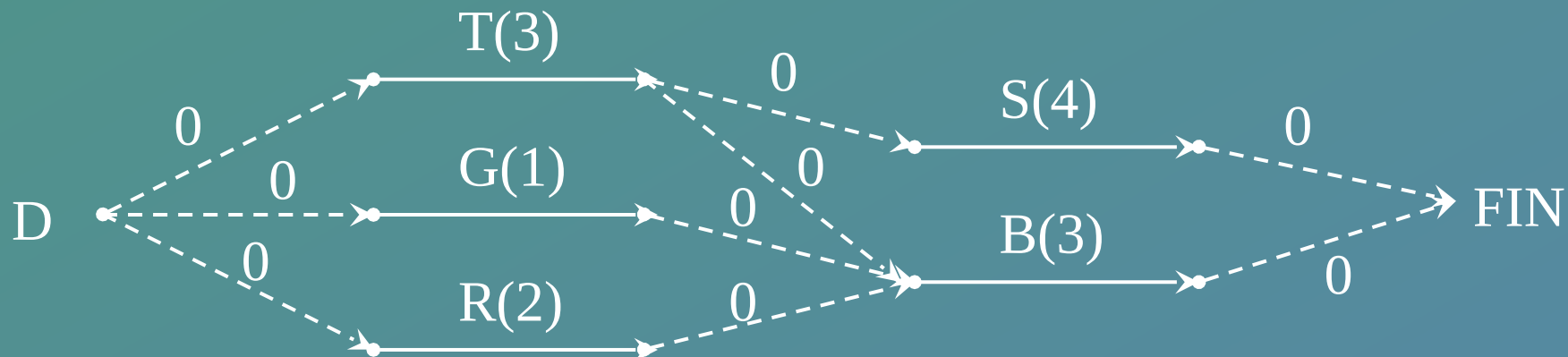


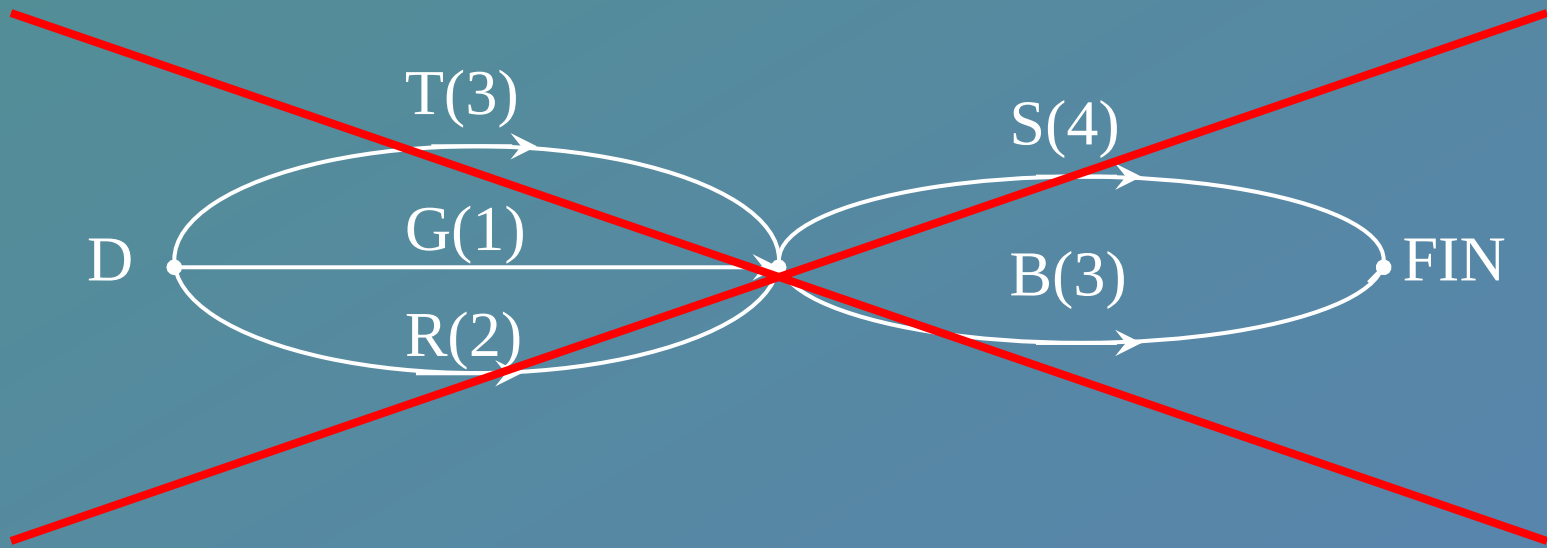
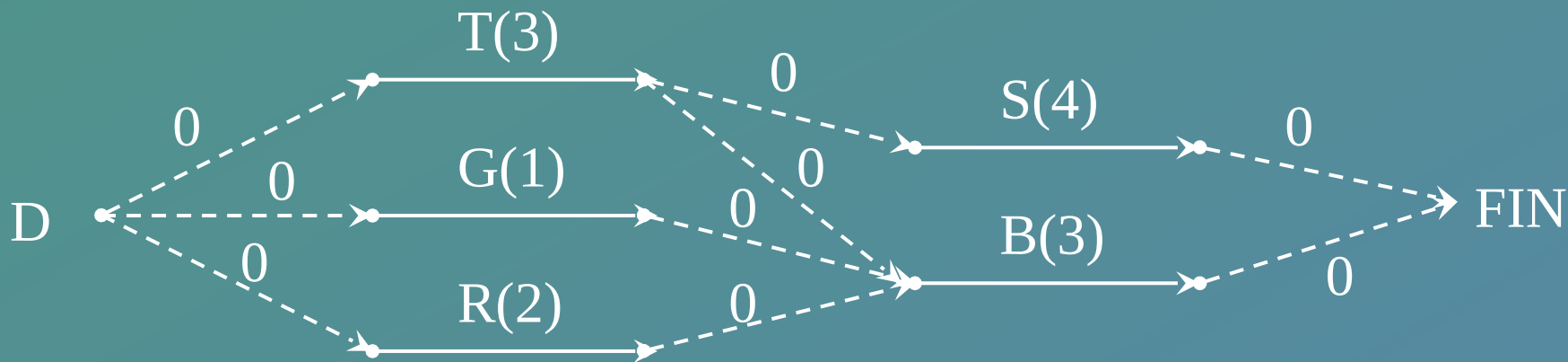








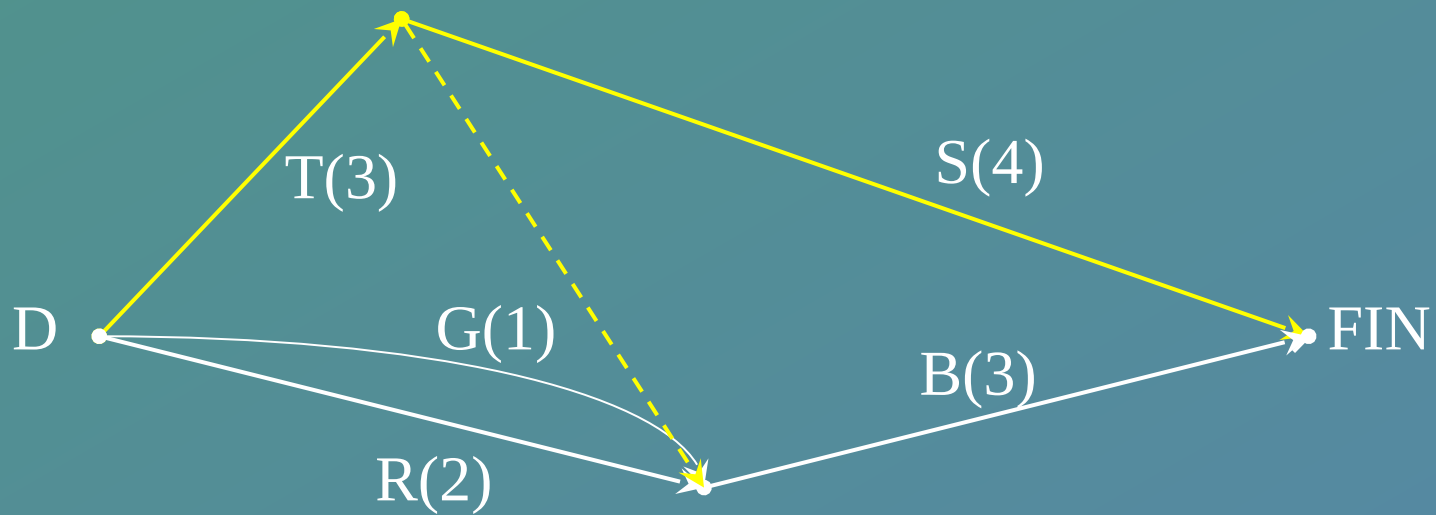


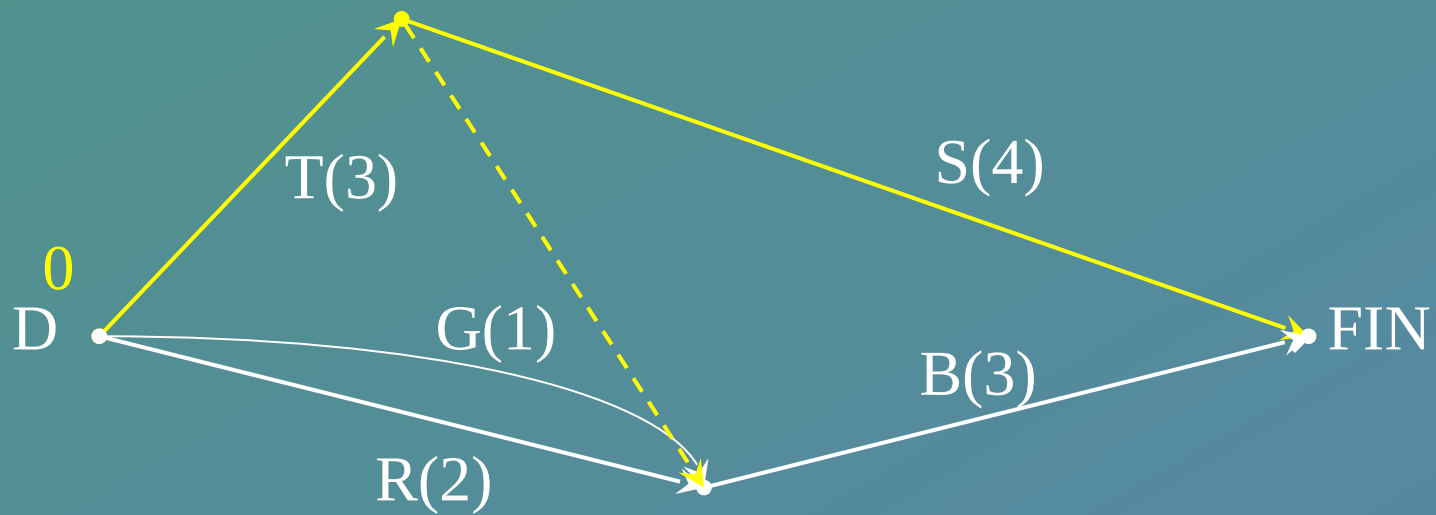


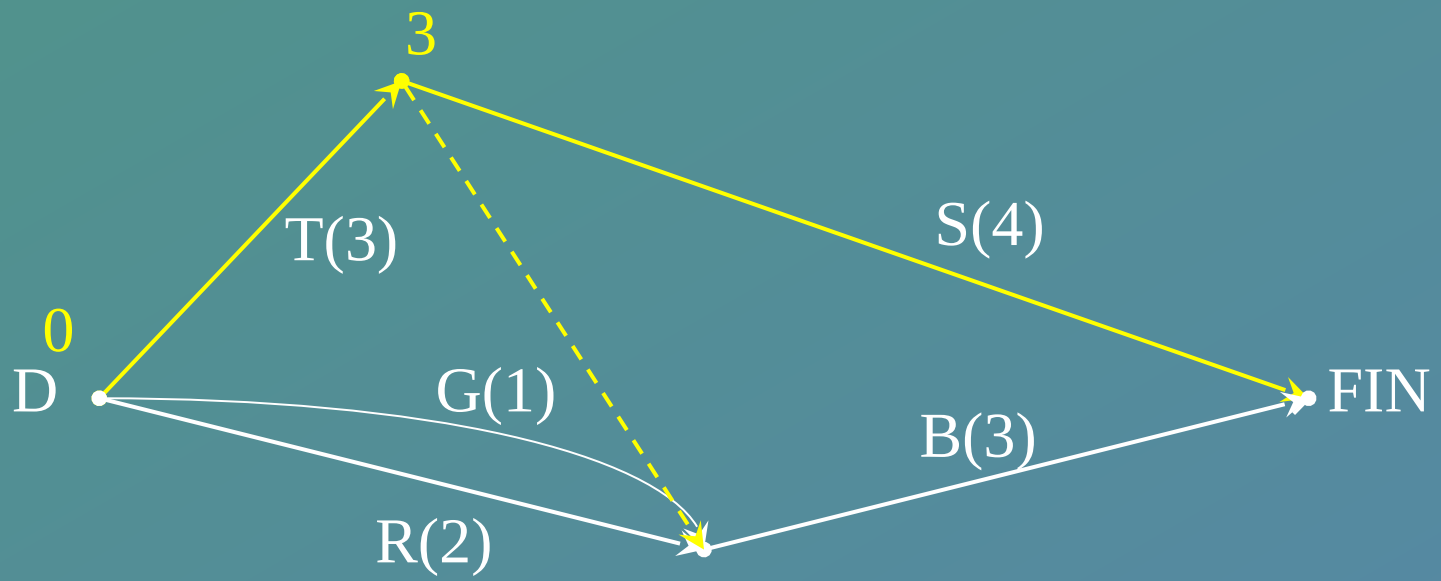
Ajoute une contrainte : **S après G**

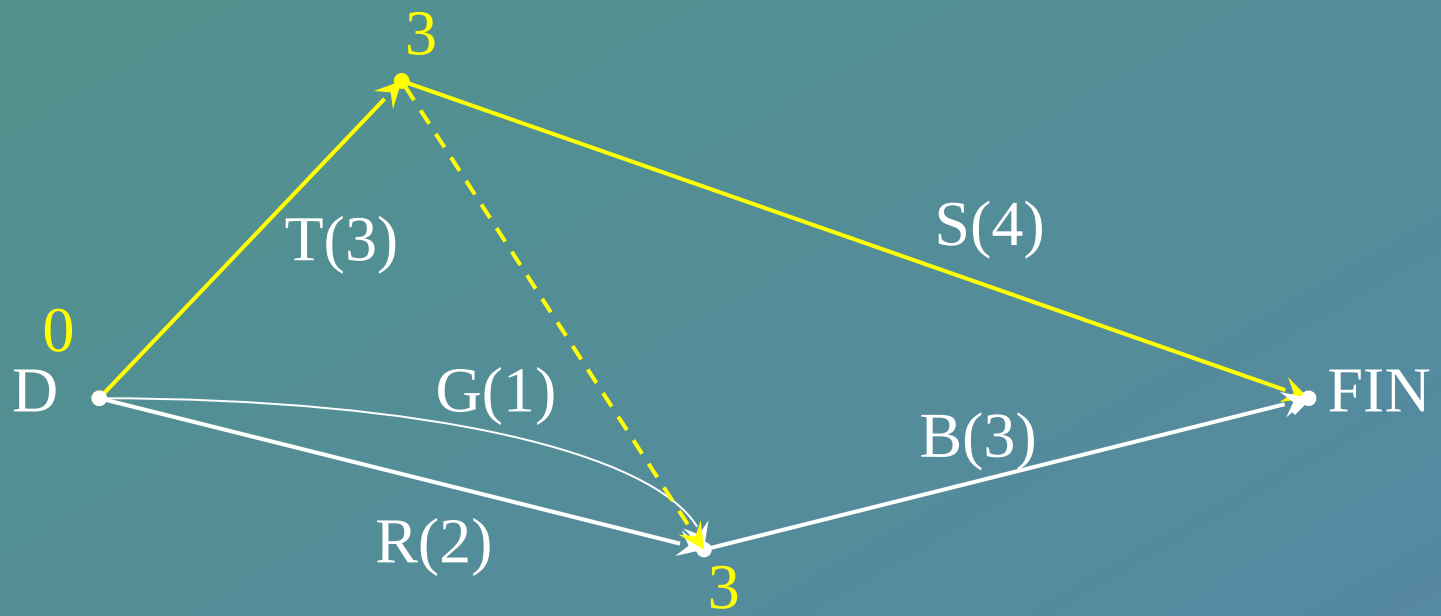


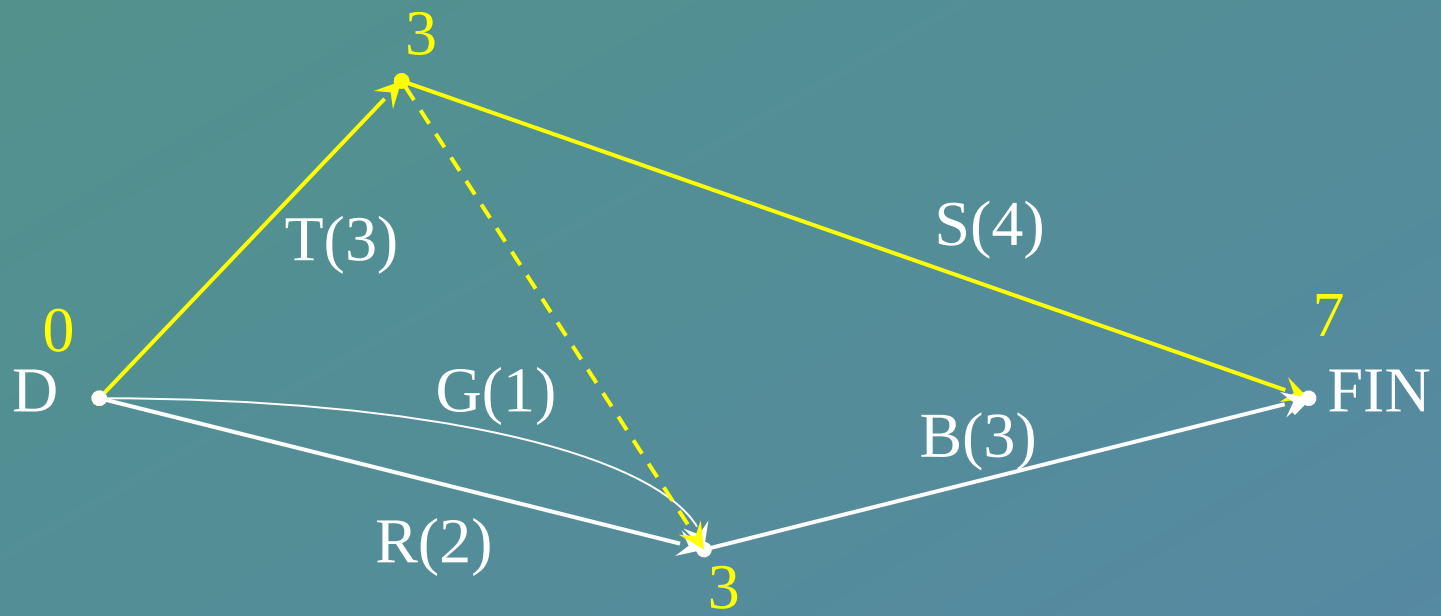


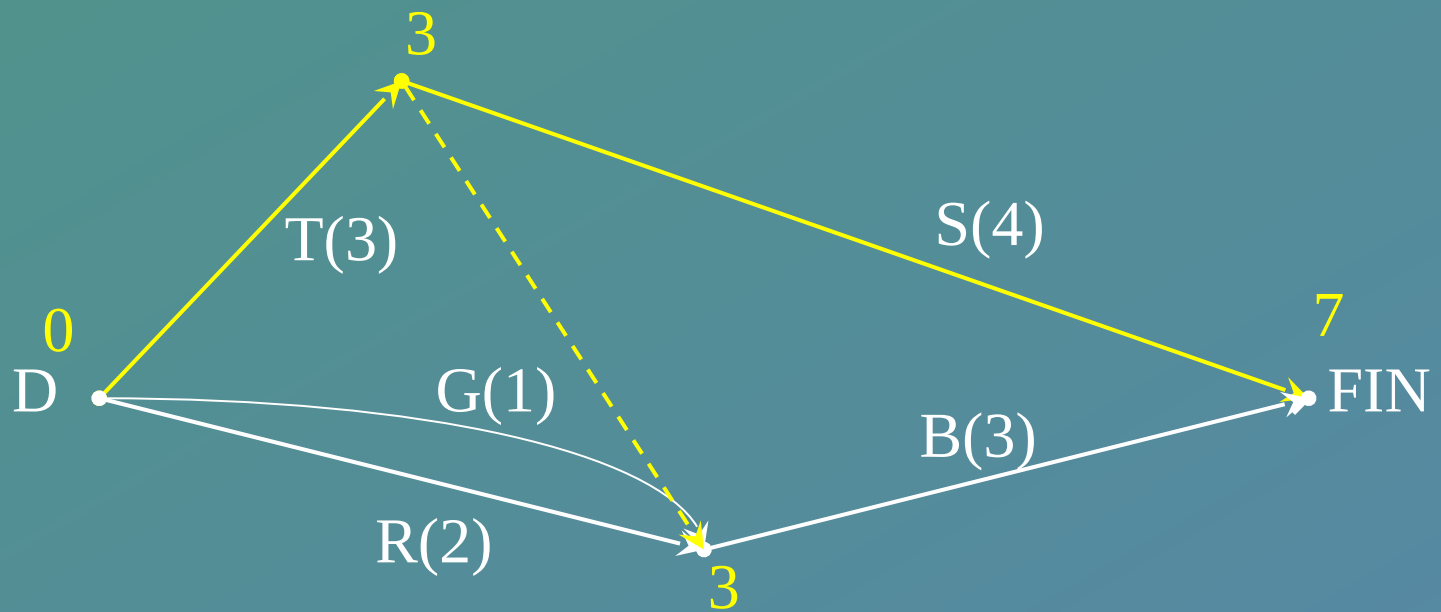








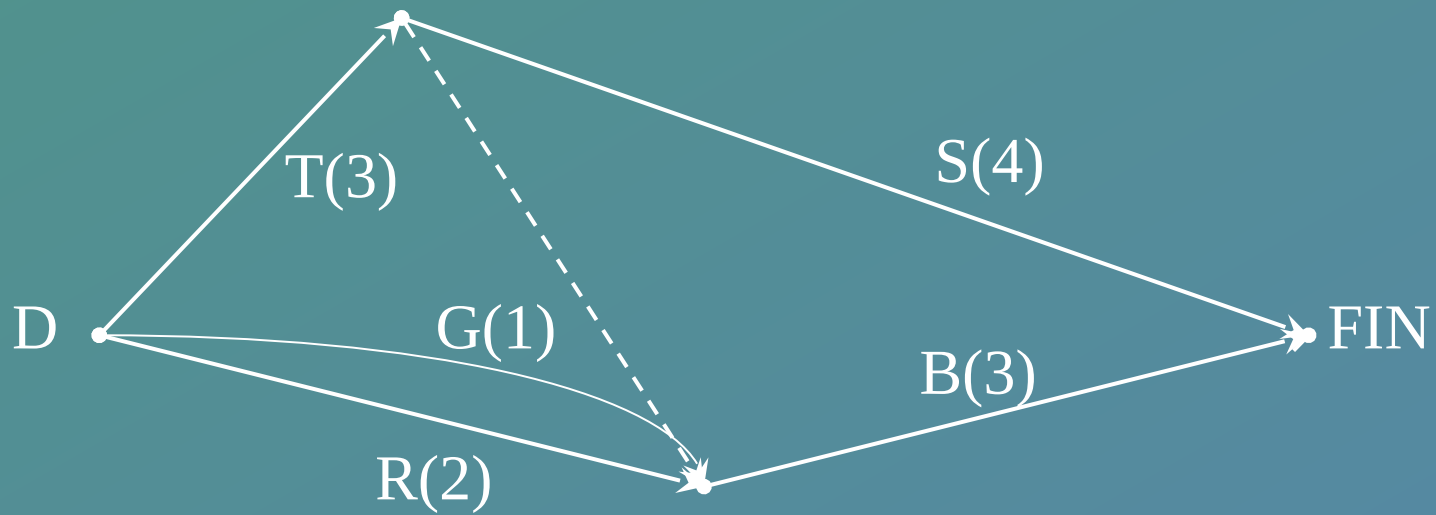


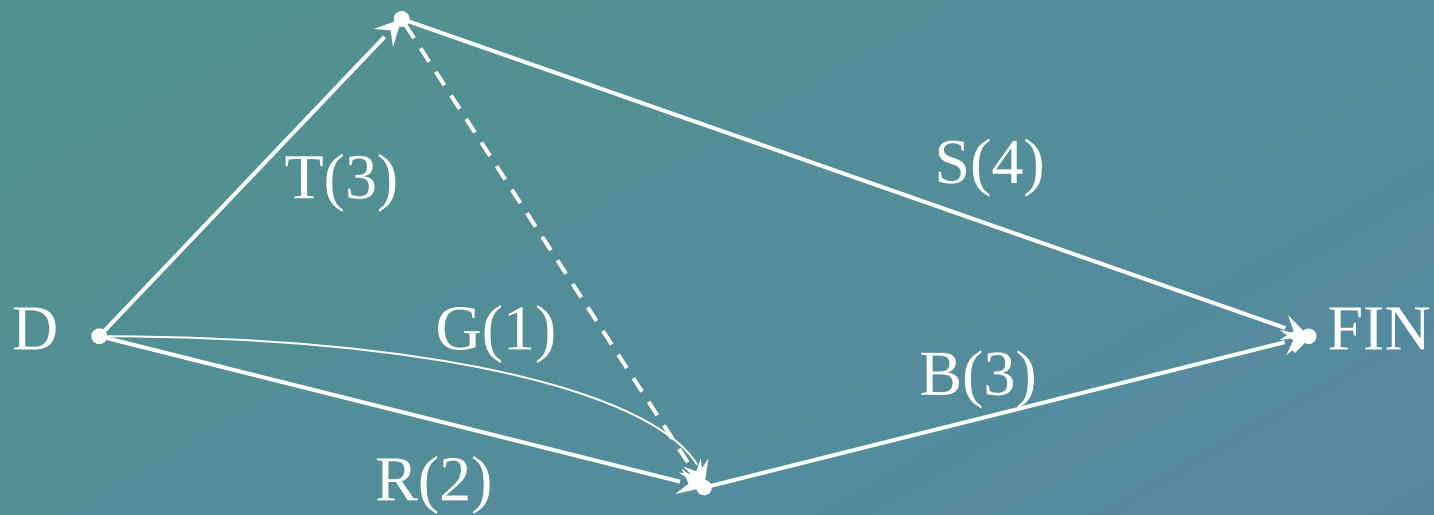


tâche	durée	après :	la date au plus tôt
Début	0	—	0
T	3	Début	0
G	1	Début	0
R	2	Début	0
B	3	T, G, R	3
S	4	T	3
Fin	0	T, G, R, B, S	7



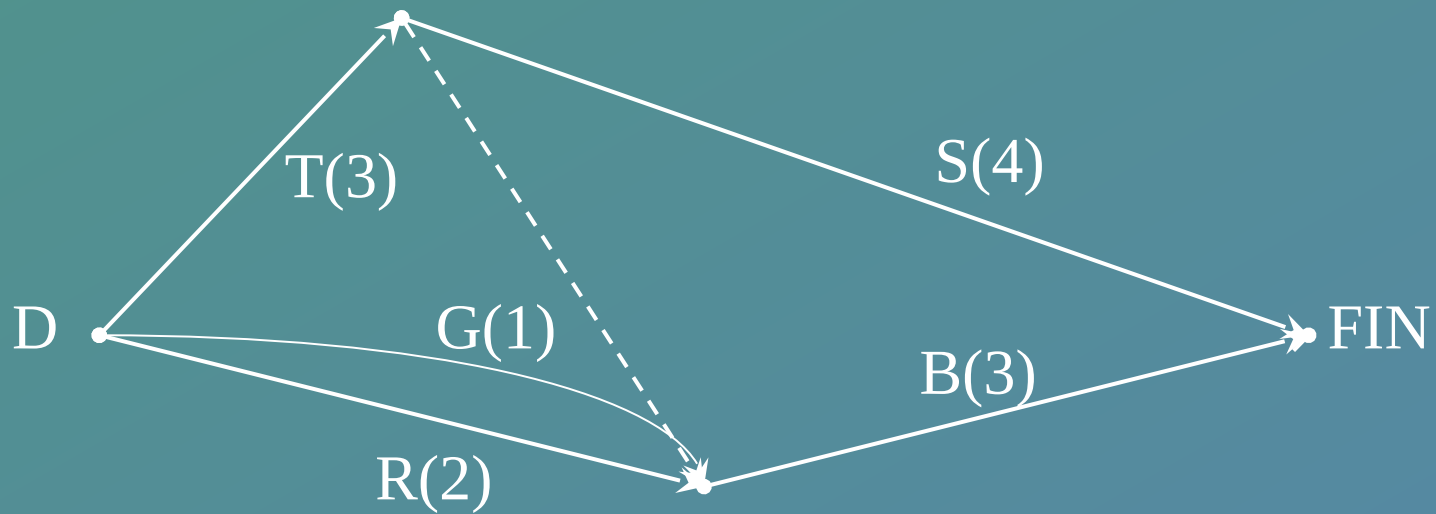






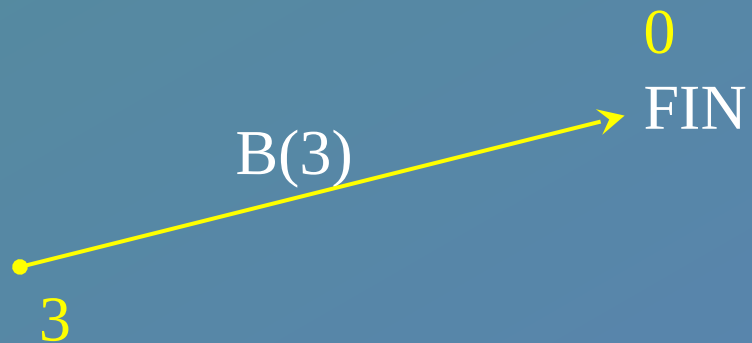
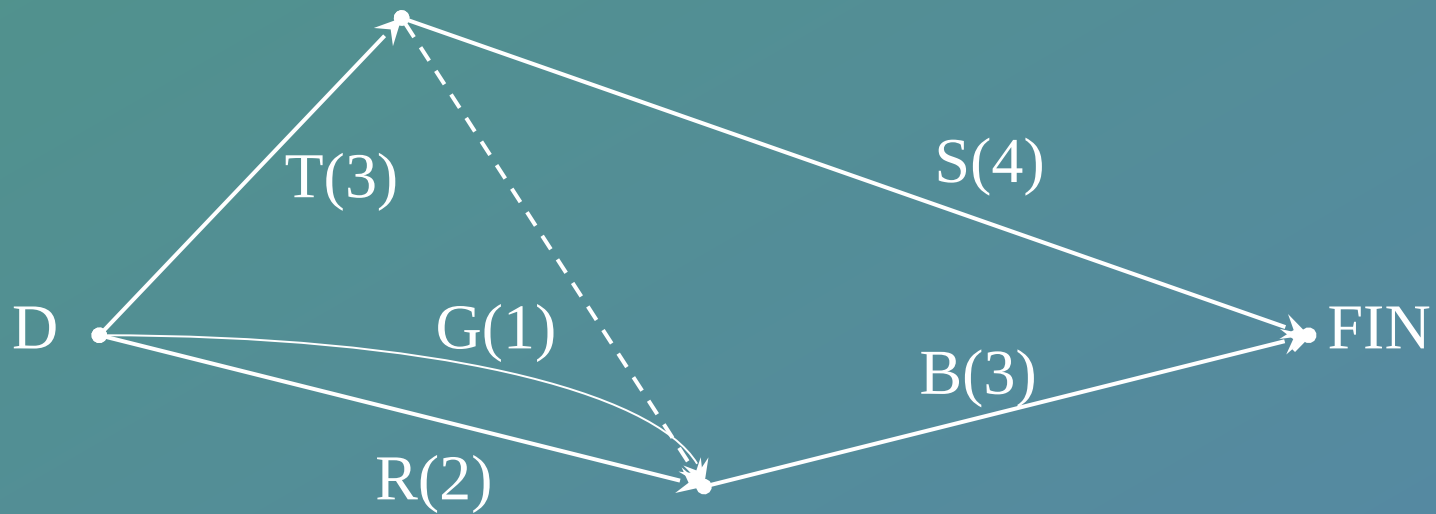
FIN

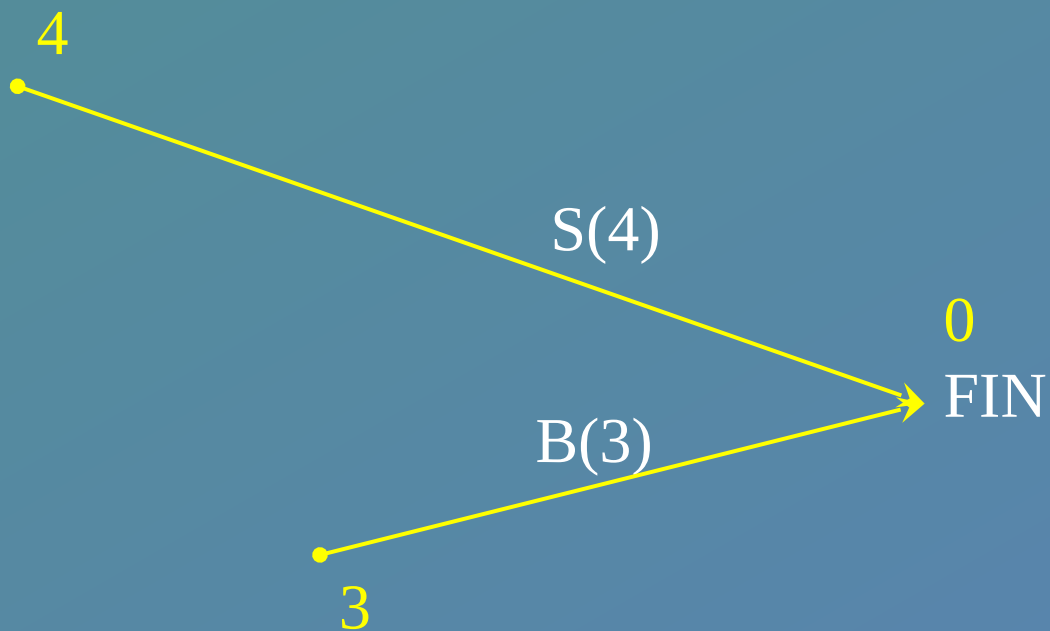
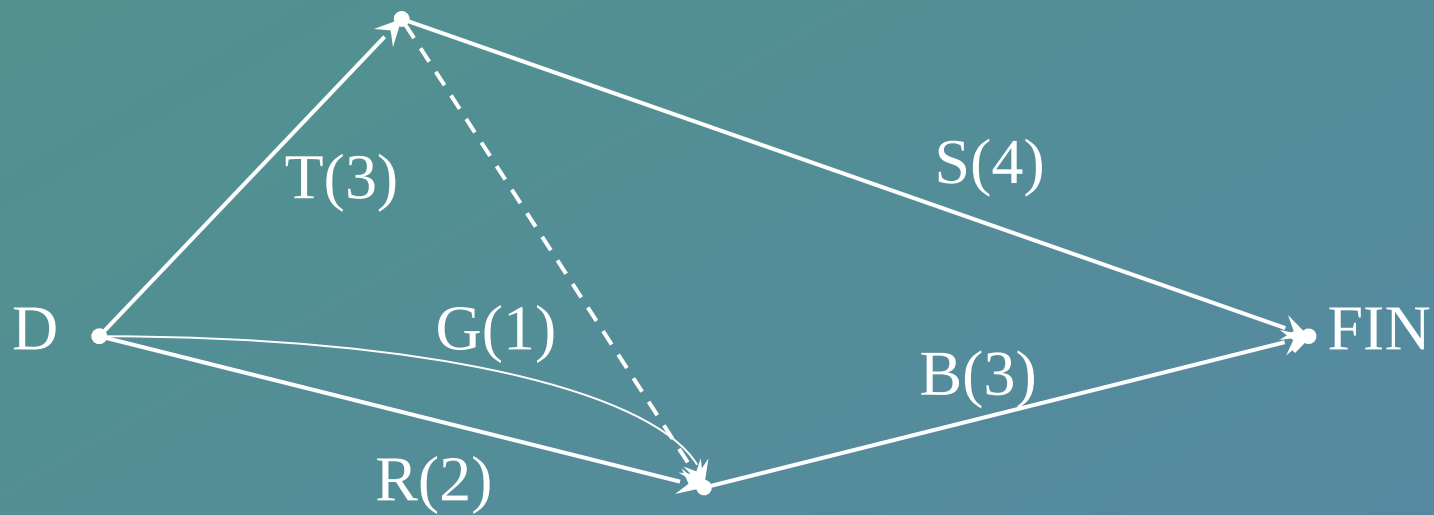


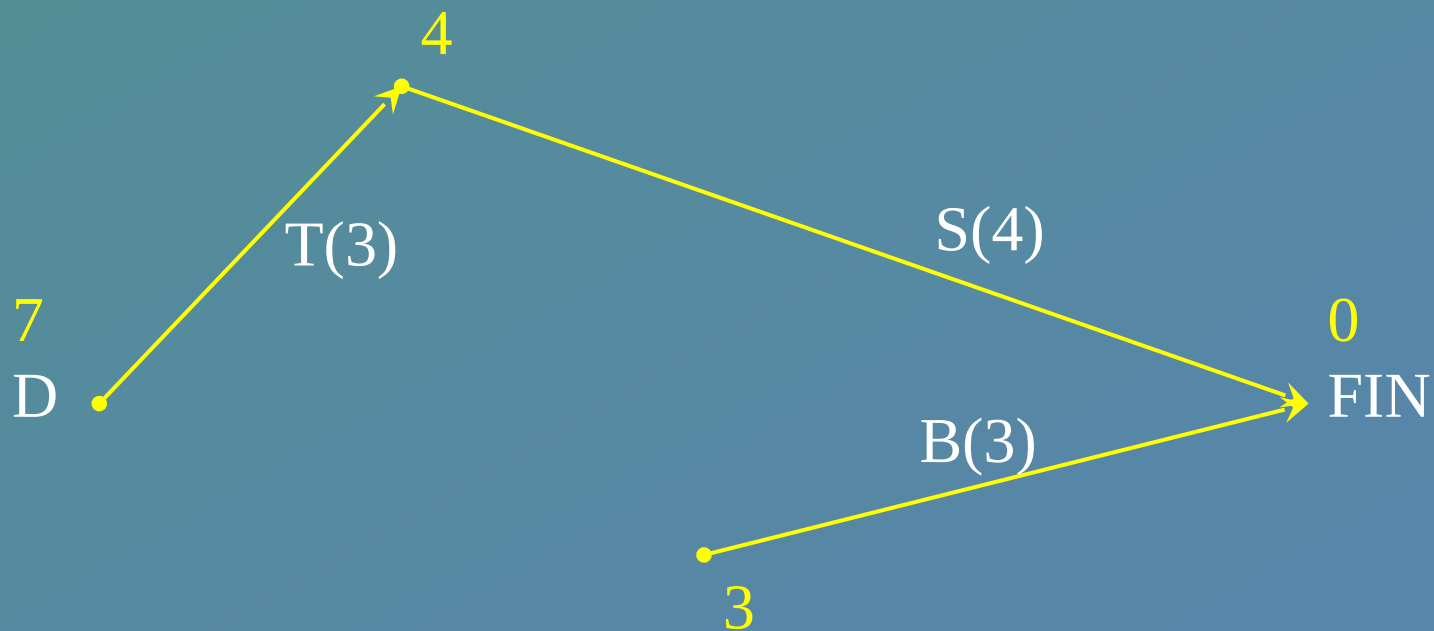
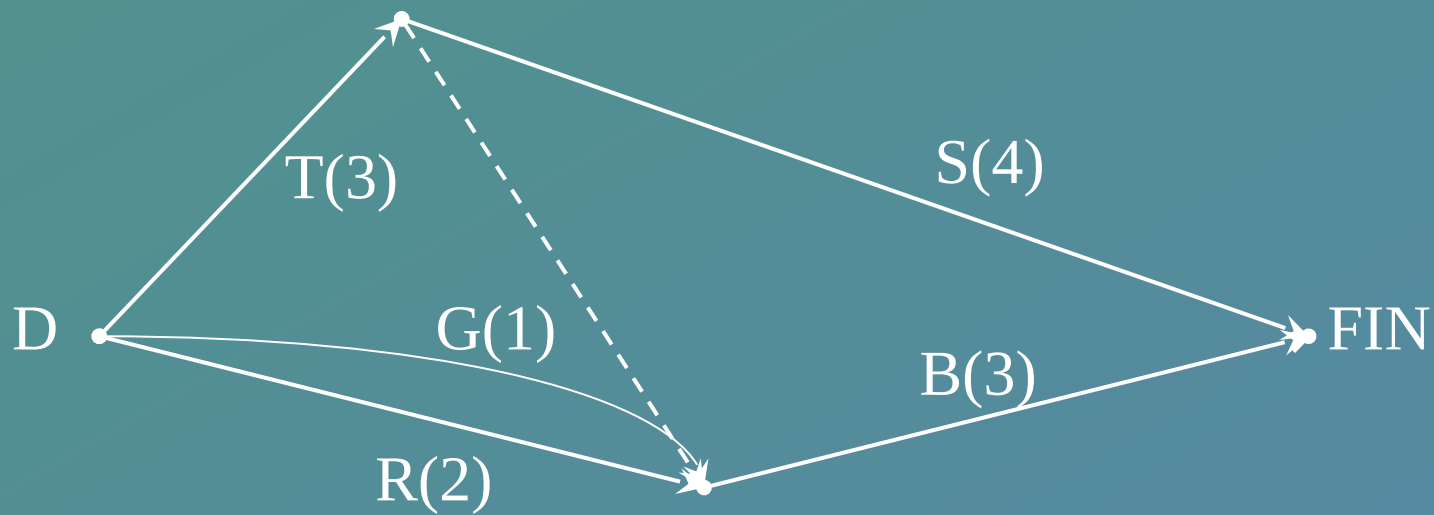


0
FIN

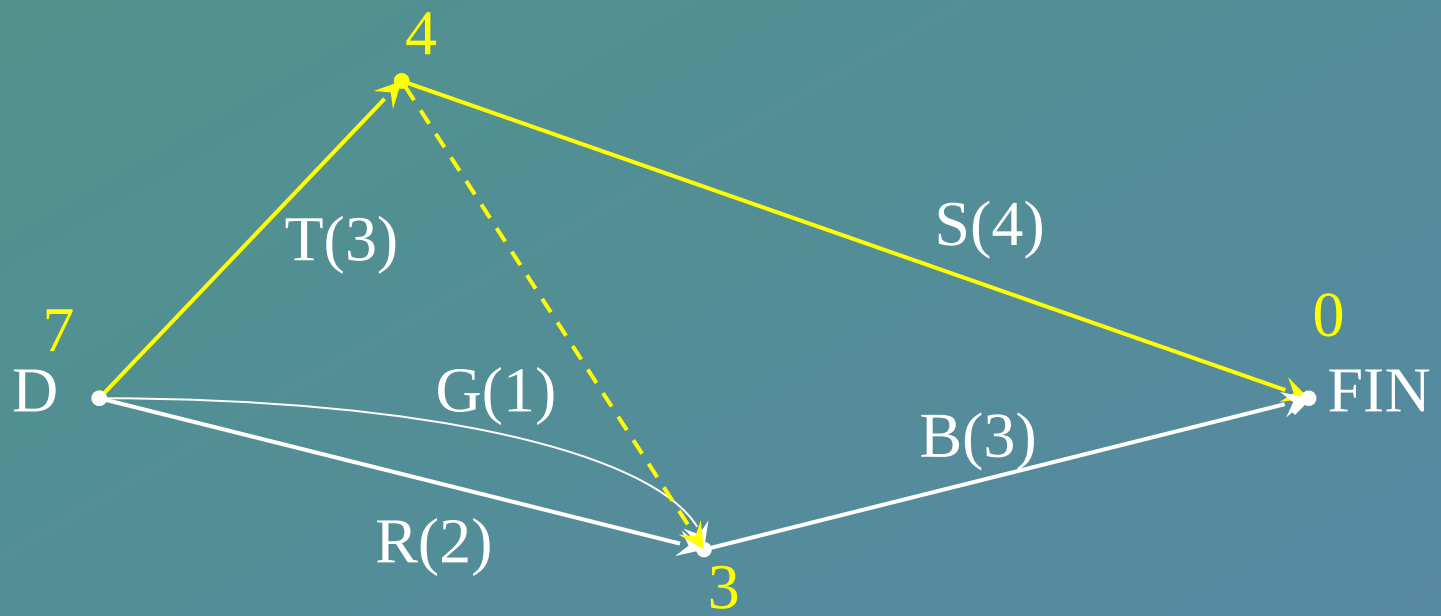


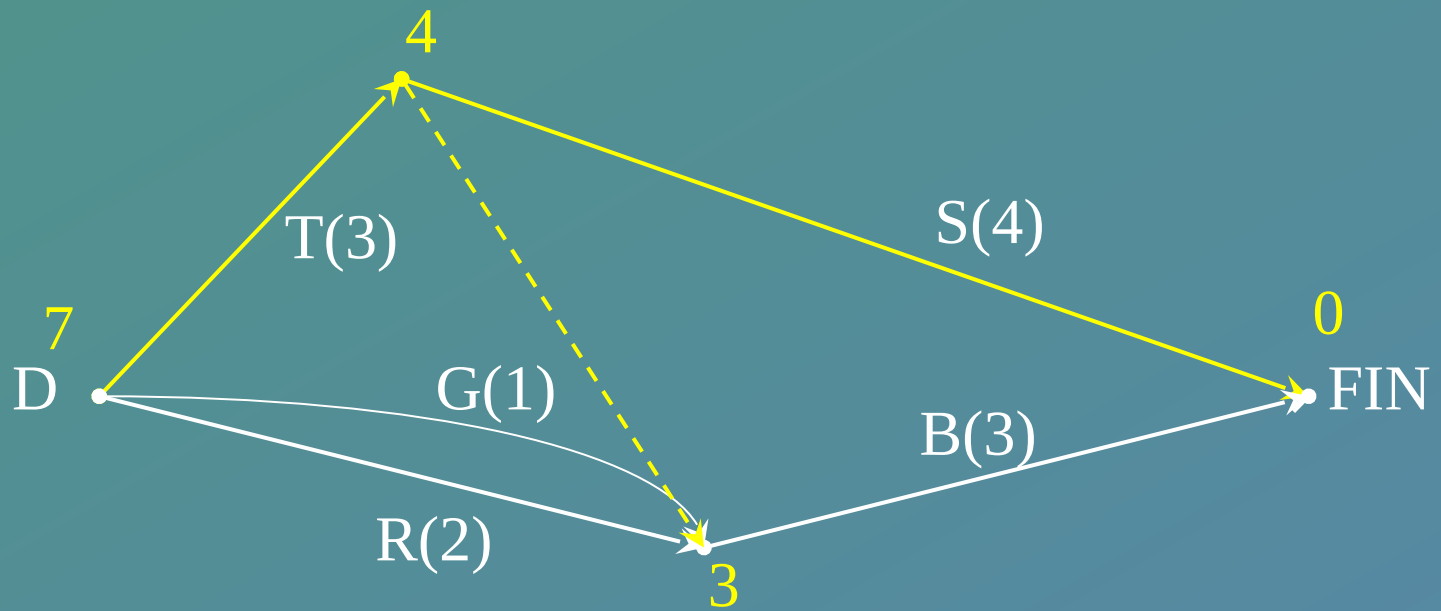






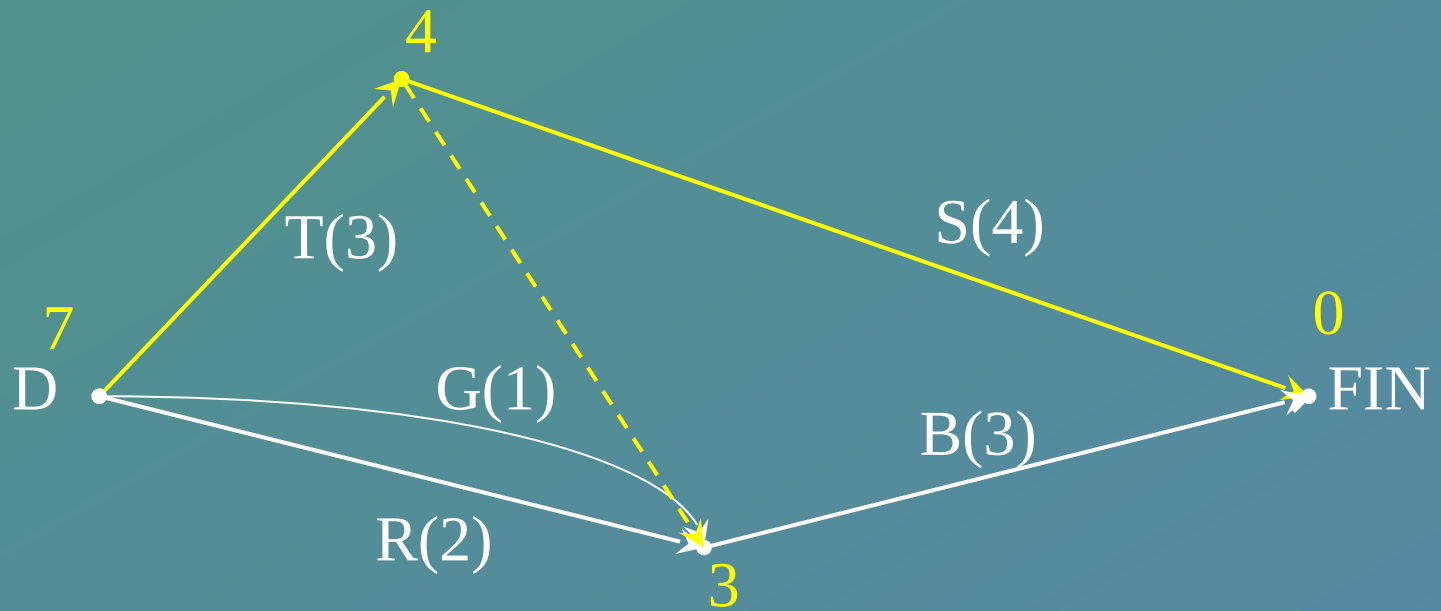






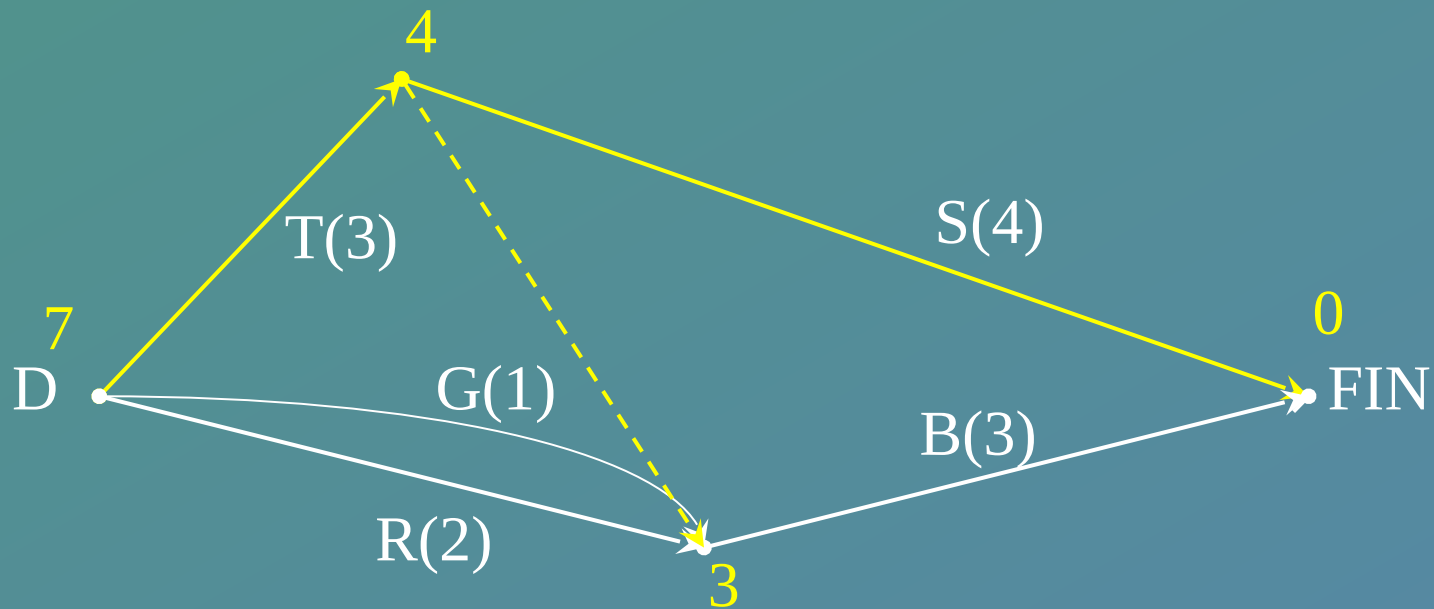
Pour une tâche, représentée par un arc $e = (x ; y)$,





Pour une tâche, représentée par un arc $e = (x ; y)$,
la date au plus tard $(e) = \pi(F) - \pi'(y) - d(e)$



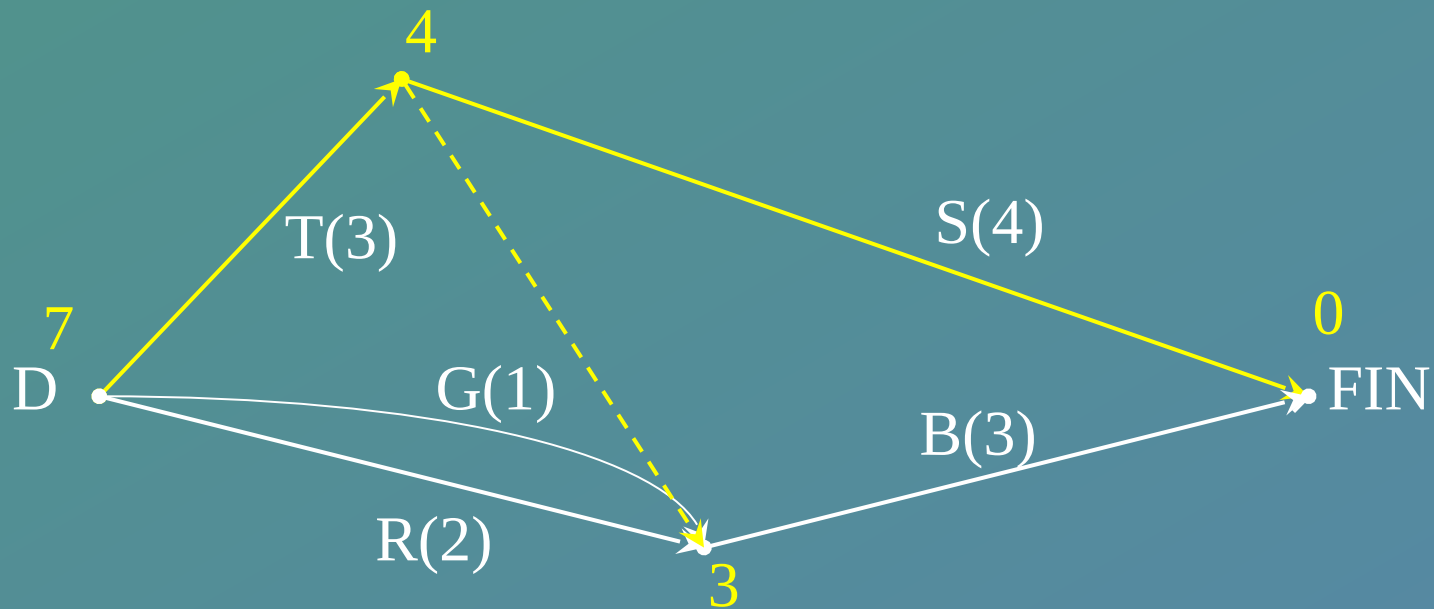


Pour une tâche, représentée par un arc $e = (x ; y)$,

la date au plus tard $(e) = \pi(F) - \pi'(y) - d(e)$

(y est le sommet terminal de e et $d(e)$ la durée de la tâche)





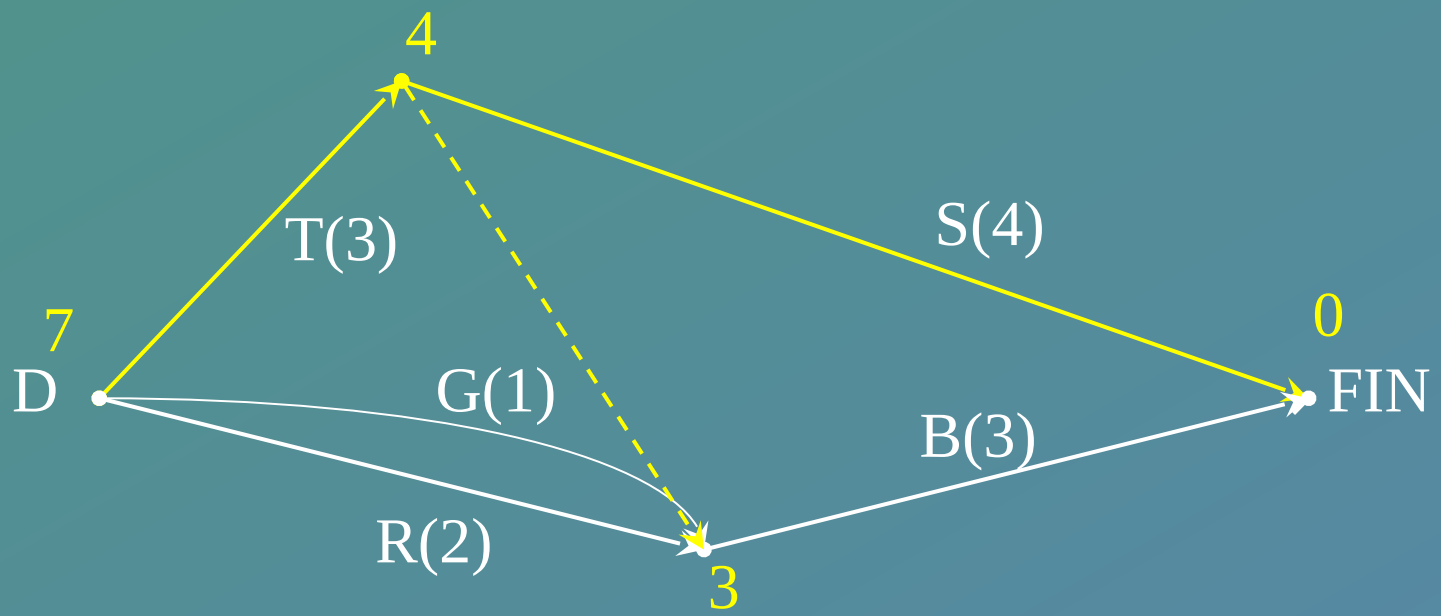
Pour une tâche, représentée par un arc $e = (x ; y)$,

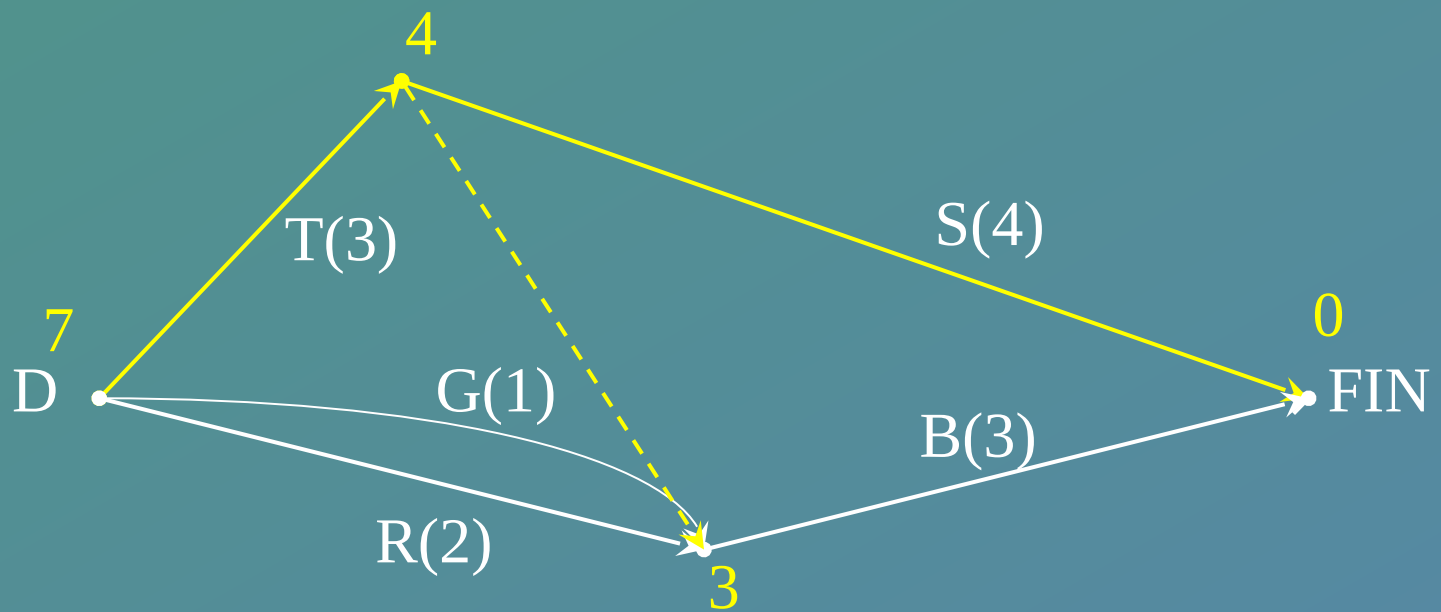
la date au plus tard (e) = $\pi(F) - \pi'(y) - d(e)$

(y est le sommet terminal de e et $d(e)$ la durée de la tâche)

la date au plus tard (G) = $7 - 3 - 1 = 3$







tâche	durée	après :	la date au plus tard
Début	0	—	0
T	3	Début	0
G	1	Début	3
R	2	Début	2
B	3	T, G, R	4
S	4	T	3
Fin	0	T, G, R, B, S	7

