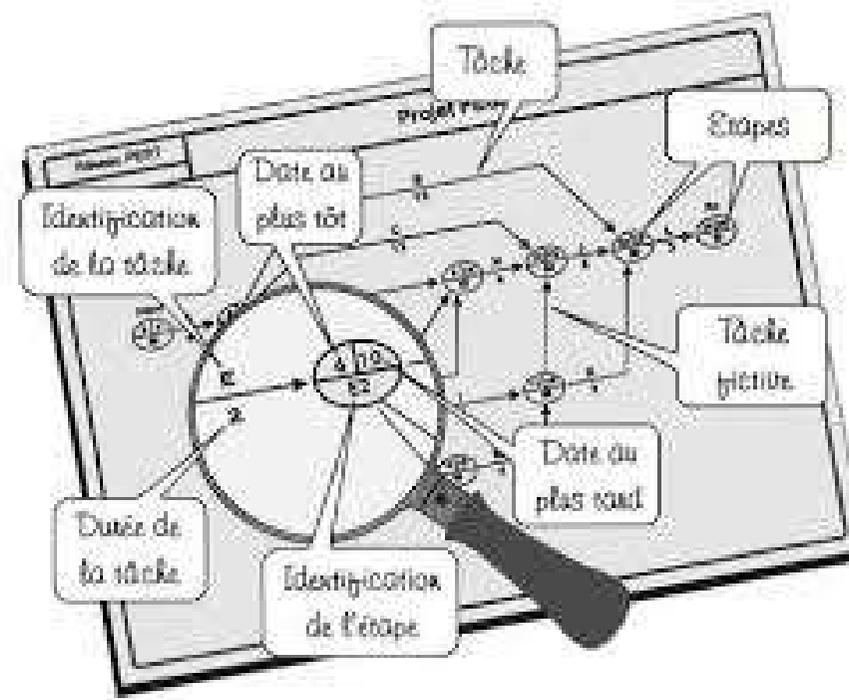


# Ordonnancement

## Méthode PERT



**Marie-Ange MANIER**

Extrait du cours MANIER M.-A. « Ordonnancement »  
(module GP85, Formation d'ingénieurs en logistique par alternance, UTBM, printemps 2021)

## *I Introduction*

### **I.1 Contexte: la gestion de projets**

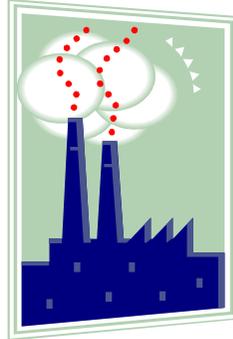
#### **Projet**

ouvrage **unitaire** composé d'un grand nombre de tâches interdépendantes, soumises à des contraintes diverses : techniques, sociales, commerciales, financières, ...

#### **Ordonnancement de projet**

déterminer un ordre d'exécution des tâches et des dates d'exécution, en respectant les contraintes imposées

- **USA** (début années 30) puis **Europe** (post 2<sup>nd</sup>e guerre mondiale)
- **origine** : problèmes posés par la réalisation d'installations industrielles (grande taille, complexité, pluridisciplinarité)



MAIS



### Projets dans l'Égypte Ancienne !!!



- \* mise en œuvre de techniques novatrices (manutention de charges lourdes,...)
- \* réunion de compétences depuis la conception jusqu'à la fin des travaux
- \* rigoureuse orchestration de toutes les disciplines

Concept de  
gestion de projet  
(project management)

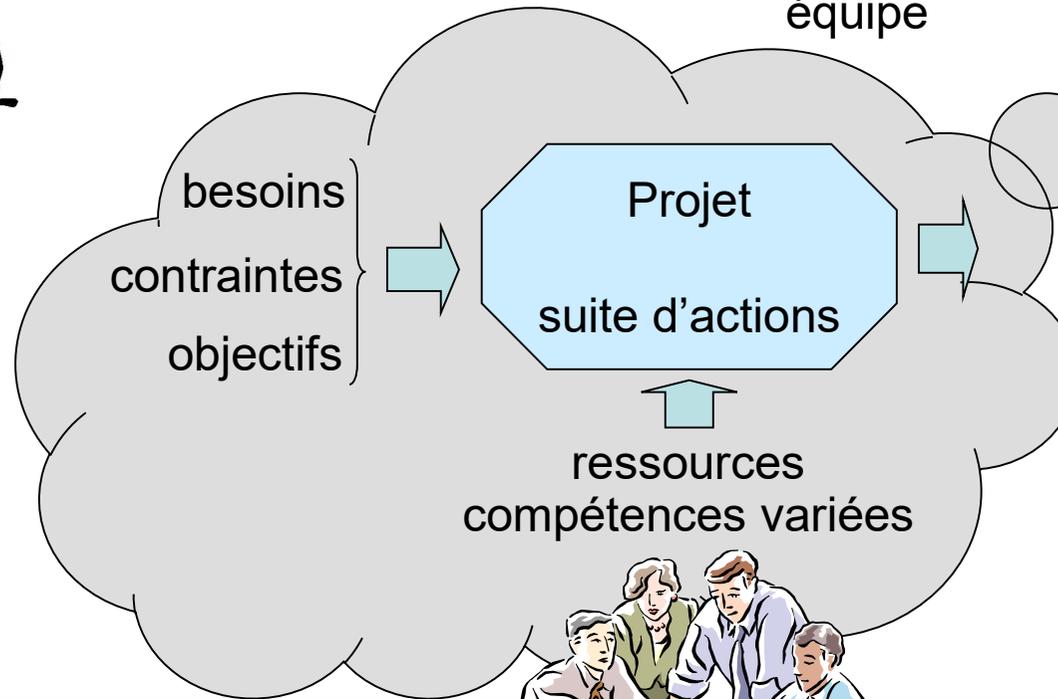
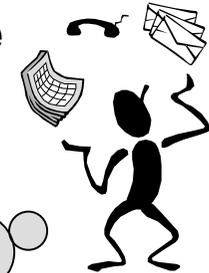
# Qu'est-ce qu'un projet ?

AFNOR : une démarche spécifique  
 ↓  
 structurer méthodiquement et progressivement une réalité à venir



**Maîtrise d'ouvrage**  
 utilisateur, client, acheteur

**Maîtrise d'œuvre**  
 chef de projet  
 équipe



Résultat précis  
 matériel /  
 intellectuel



## Caractéristiques d'un projet



- action ponctuelle, unique, non répétitive, novatrice



limitée dans le temps :

date de début, date de fin, **planification**



qui comporte une part de risques

**estimation**



est contrainte par des **objectifs** :- **qualité**

- **coûts**

- **délais**



mobilise des compétences complémentaires (travail d'équipe)

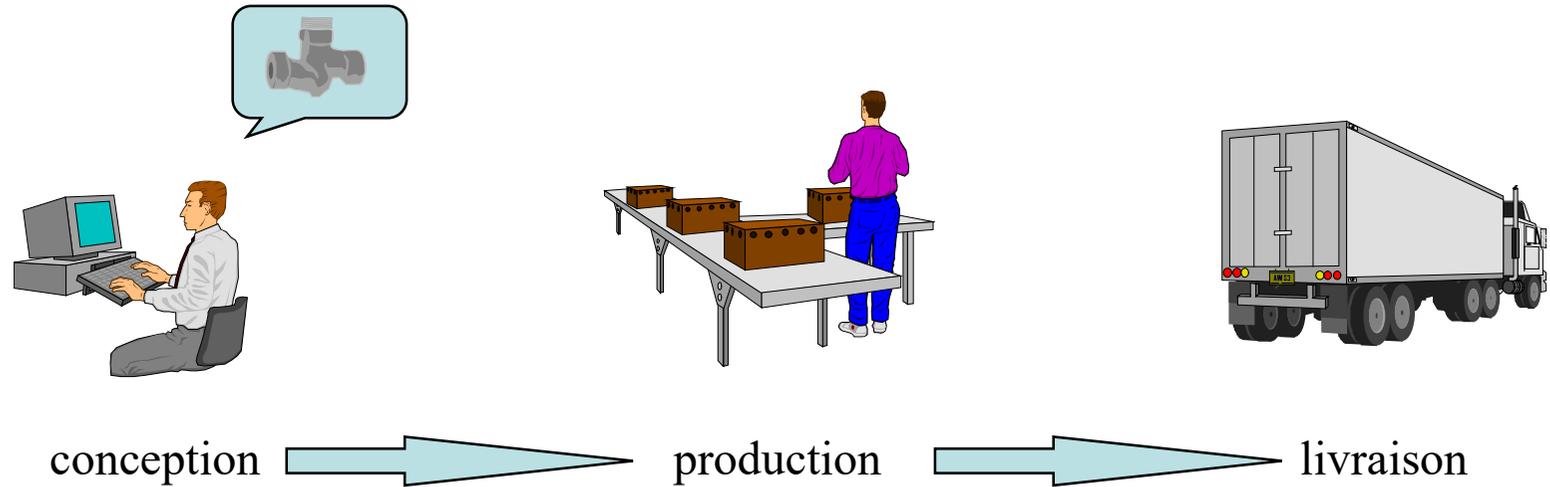


doit faire l'objet d'un suivi et d'un pilotage rigoureux

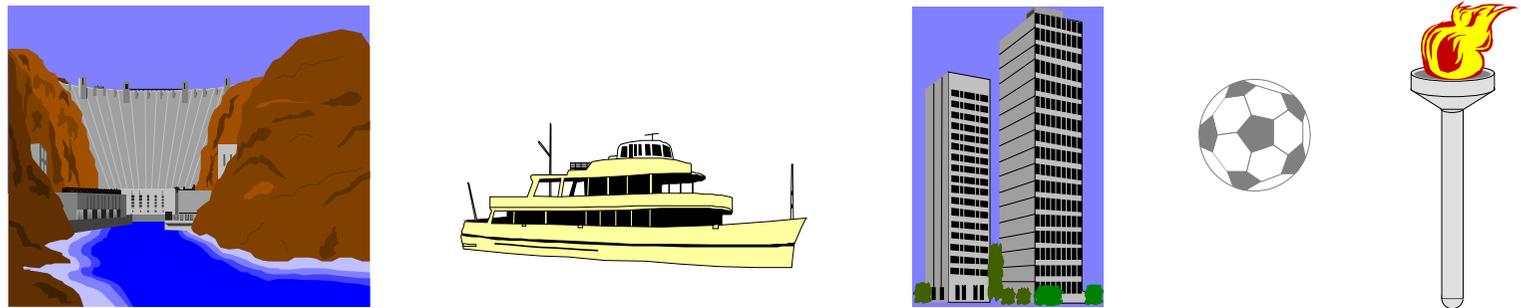
( étapes, jalons, points de contrôle)

# Projet : exemples d'application

- domaine industriel



- autres



- domaine informatique



- Développement d'une nouvelle application informatique
- Evolution d'une application (décision interne / réglementation)
- Réalisation d'un outil de communication externe ou interne
- Définition/mise en place d'une nouvelle organisation du travail
- Acquisition/mise en place d'un outil, de matériel (besoins spécifiques)...
- **Une procédure**
- **Une activité permanente de l'entreprise**
- **Le produit attendu**

**ne sont pas  
des projets**

## Planification de projet

=

réalisation chronologique des actions suivantes:

- découper le projet en activités élémentaires
- identifier les contraintes
- modéliser l'enchaînement des activités
- évaluer les durées
- établir la planification
- déterminer les latitudes d'action
- optimiser la planification

**ordonnancement  
de projet**

# *I Introduction*

## **I.2 Définitions**

### \* **Ordonnancement de projet :**

Déterminer un **ordre d'exécution** des tâches qui ont été identifiées et des **dates** d'exécution en respectant les **contraintes** imposées

### \* **Les données d'un problème d'ordonnancement**

- tâches

- ressources

- contraintes : \* temporelles (délai, contraintes d'enchaînement,...)  
\* sur l'utilisation et la disponibilité des **ressources**.

- le ou les objectifs

*relations liant  
tâches et ressources*

**Les objectifs** : résoudre un problème d'ordonnancement, c'est trouver, parmi l'ensemble des solutions, celle qui optimise un ou plusieurs critères

=> **ORDONNANCEMENT OPTIMAL**

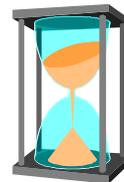
Critères à minimiser en ordonnancement de projet :



**durée totale** d'exécution du projet



**coût** d'exécution du projet



ET



Exemple: réalisation d'un support, constituant un des éléments du projet Mididanse.

Tableau des tâches à effectuer et de leurs contraintes d'antériorité:

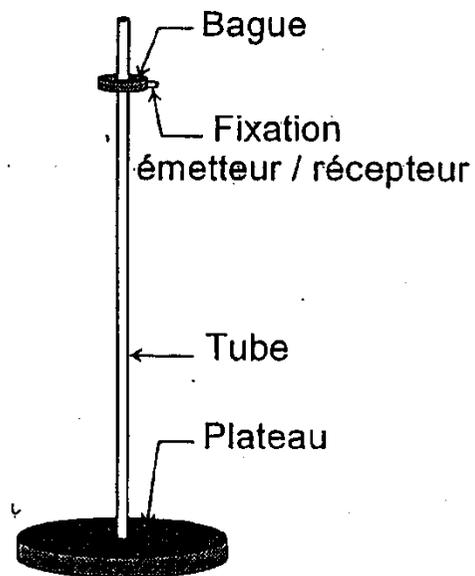


Figure – Support pour émetteur / récepteur infrarouge

activités	code	activités antérieures
concevoir le support	E	début
approvisionner la matière	N	E
découper les tubes	O	N
usiner les bagues	P	N
sous-traiter la découpe des plateaux	Q	N
usiner les tubes	R	O
usiner les plateaux	S	Q
anodiser	T	R, P, S
assembler	Y	T

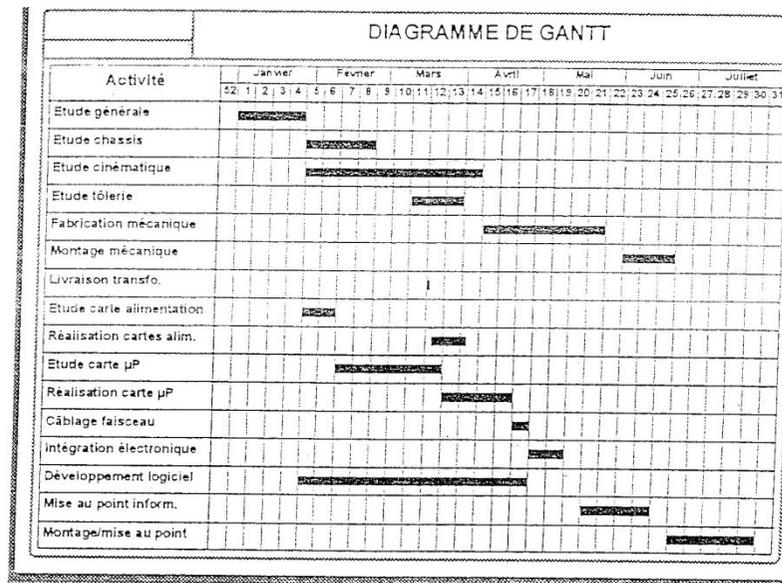
## Exemple: projet Mididanse

Sous ensemble	Activité	Code	Durée	Antériorité
Conception générale		A	1j	Début
Carte processeur	Conception	B	7j	A
« «	Commande composants	F	10j	B
« «	Etude circuit imprimé	G	3j	B
« «	Réalisation circuit imprimé	H	2j	G
« «	Câblage	U	2j	F,H
« «	Programmation	V	3j	U
« «	Mise au point	W	1j	V
Procédé infrarouge	Conception	C	7j	A
« «	Commande composants	I	9j	C
« «	Dessin circuit imprimé	J	5j	C
« «	Câblage / mise au point	K	5j	I,J
Procédé HF	Conception	D	7j	A
« «	Dessin circuit imprimé	L	3j	D
« «	Câblage	M	5j	<del>M</del> L
« «	Mise au point / contrôles	X	3j	<del>X</del> M
Supports	Concevoir le support	E	5j	A
« «	Approvisionner matière	N	10j	E
« «	Découper les tubes	O	1j	N
« «	Usiner les bagues	P	2j	N
« «	Sous-traiter la découpe des plateaux	Q	5j	N
« «	Usiner les tubes	R	1j	O
« «	Usiner les plateaux	S	3j	Q
« «	Anodiser	T	5j	R,P,S
« «	Assembler	Y	1j	T
Intégration		Z	3j	K,W,X,Y

## I.3 Méthodes

### \* Jusqu'en 1957: **diagramme de GANTT**

Construction: tenir compte à la fois de la durée et des dates d'exécution (contraintes potentielles), et des ressources (contraintes disjonctives et cumulatives)



- ⇒ complexe pour plus de 30 tâches
- ⇒ inconvénient majeur : toute modification du planning implique de reconstruire totalement le graphique
- ⇒ limitation du nombre de mises à jour

### I.3 Méthodes

\* Quelques algorithmes de résolution disponibles pour les problèmes d'ordonnancement (exe: algorithme de Johnson)

#### MAIS

Cas particuliers (2 ou 3 postes successifs)

\* Méthodes non adaptées au cas de fabrications

- non répétitives
- très complexes
- grand nombre de données, à spécifications très diverses
- nature des exigences associées: technologique, financière, sociale, climatique, commerciale, internationale....

=> Nécessite une **coordination** poussée de l'ensemble des moyens à mettre en oeuvre

\* 1958: méthodes fondées sur la **Théorie des Graphes**

**USA**

**P.E.R.T.** : Program Evaluation and Review Technic

Créateur : armée USA

Application : Projet militaire : fusées Polaris

Résultats : durée du projet : 5 ans au lieu de 7

**France**

**M.P.M.** : Méthode des Potentiels Metra

Créateur : Bernard Roy : thèse de Mathématiques  
Appliquées

Application : Construction du paquebot France

**GB**

**C.P.M.** : Critical Method Path

Créateur : Cie anglaise d'électricité

Application : vérification des usines électriques

Résultats : Gain de temps de 30%

**Objectifs:**

- établir la suite des tâches d'un projet et de représenter **graphiquement** l'enchaînement de ces tâches (**contraintes potentielles**)
- d 'abord*
- déterminer le **meilleur temps total** nécessaire à la réalisation de l'objectif
  - déterminer les **tâches critiques**, celles dont l'exécution ne peut être retardée ni ralentie, sans que la fin du projet ne soit décalée du temps correspondant.
- puis*
- **évaluer** les moyens à mettre en oeuvre
  - connaître les **conséquences** du changement de durée d'une tâche du projet.
- enfin*
- obtenir le plus **bas coût** possible  
(aménagement de certaines tâches)
  - **raccourcir** le programme initial au **moindre coût**  
(raccourcir le temps d'exécution de certaines tâches).

## II La méthode PERT

Version initiale du PERT : on ne tient compte que des **contraintes potentielles**  
(pas des contraintes de ressources)

**PERT = graphe potentiels-événements**  
**graphe potentiels-étapes**

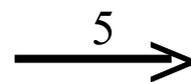
### II.1 Définitions et conventions

PERT : graphe **orienté** dont les **sommets** représentant des étapes dans le déroulement du projet  
et dont les **arcs** sont associés aux tâches à effectuer



**étape** ou évènement  
=  
début ou fin d'une tâche

(10 : identificateur)

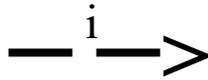


**tâche** (opération)  
valeur de l'arc  
= durée de la tâche

**Etape: de durée nulle**  
=> **ne coûte rien**

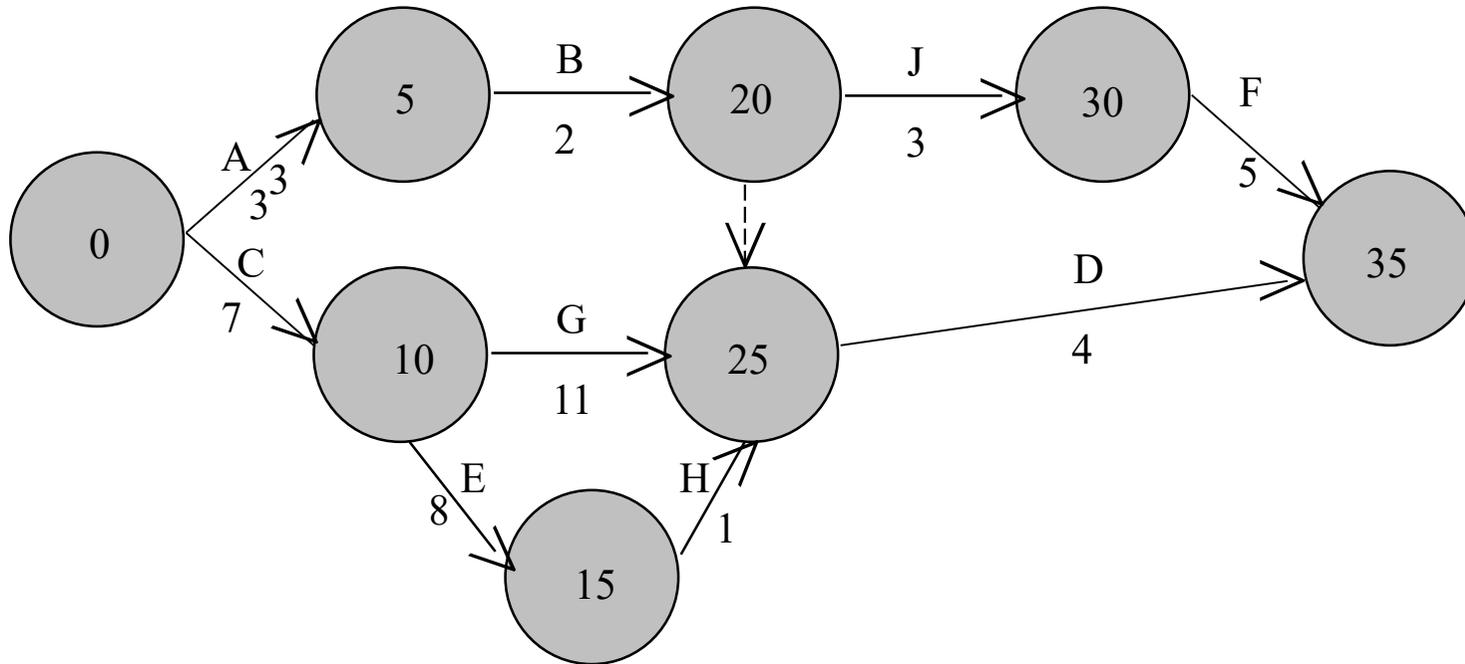
**Tâche: fait évoluer le projet vers son état final.**  
=> **consomme temps**, énergie, matière, etc.  
=> on lui associe un **coût**

+



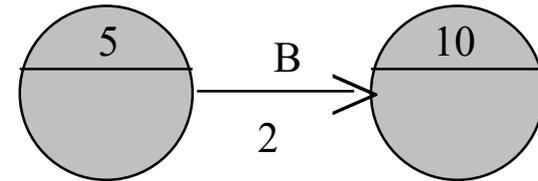
Une **tâche fictive** (i) représente une **contrainte** entre tâches non indépendantes. On lui associe un **arc en pointillés**. Sa durée est nulle, elle ne consomme aucune ressource : elle ne coûte rien.

Exemple:



# Conventions

- toute tâche a **une** étape de début et **une** étape de fin



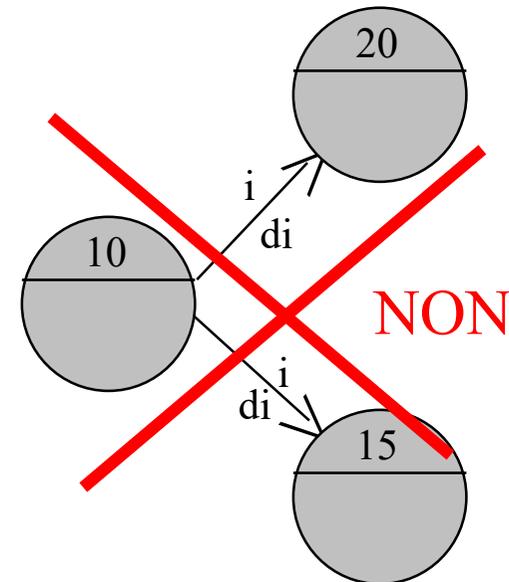
---

- un graphe commence et termine **toujours par un seul sommet**

sommets de début et fin du graphe : étapes de début et fin du projet

---

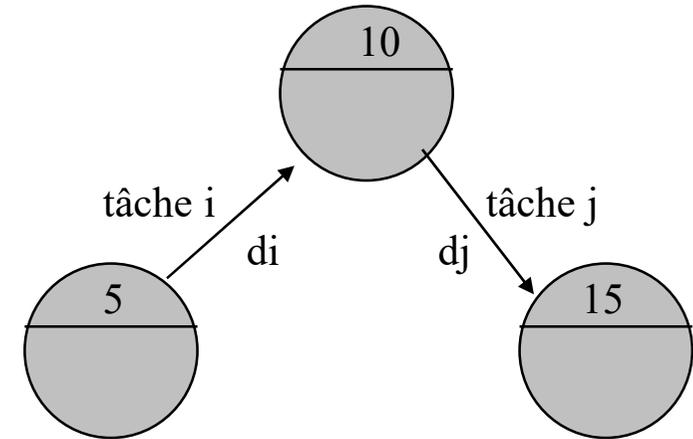
- on ne peut représenter une tâche que par **un seul arc**



- une tâche A est **antérieure** à une tâche B si A doit être réalisée avant B
- les antériorités immédiates sont appelées **antécédences**

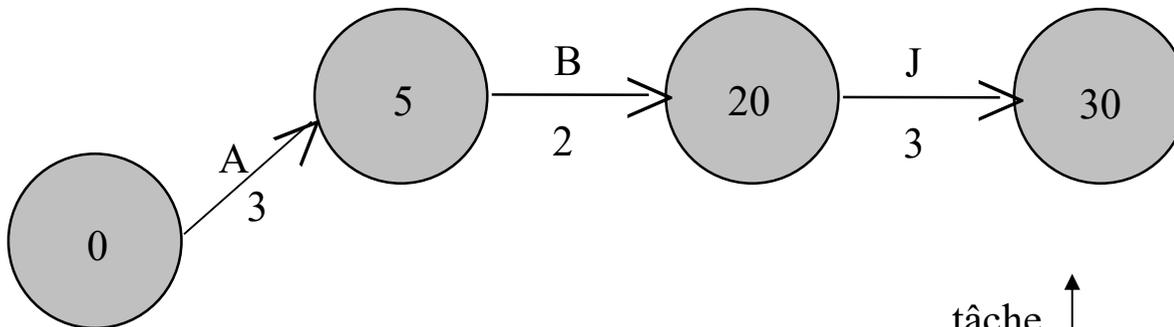
\* Exemple 1:

2 tâches i et j de durées respectives  $d_i$  et  $d_j$ , avec la contrainte "i est antérieure à j"

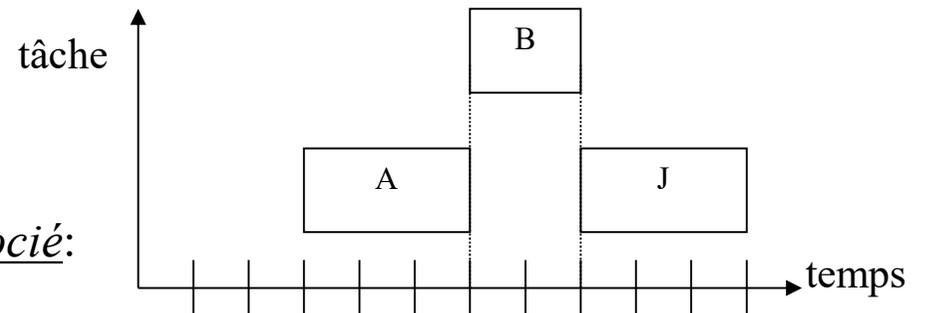


\* Exemple 2:

A et B sont **antérieures** à J  
B est **antécédente** de J



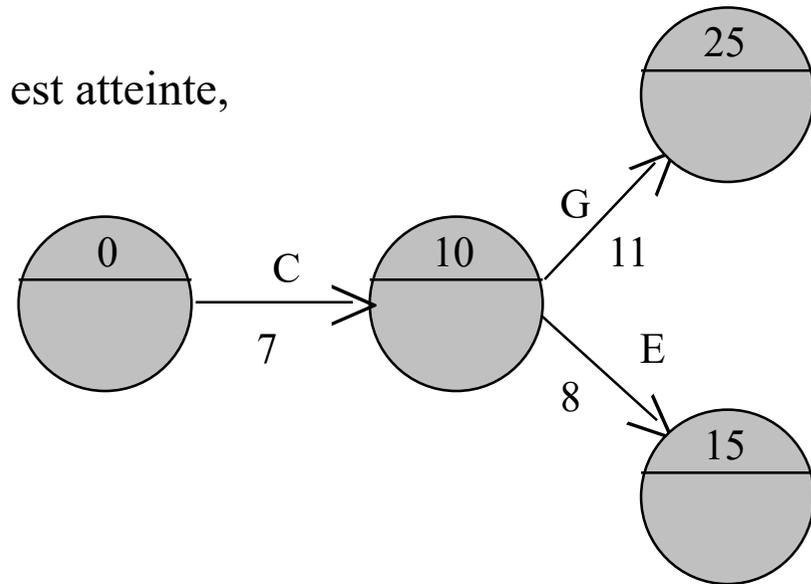
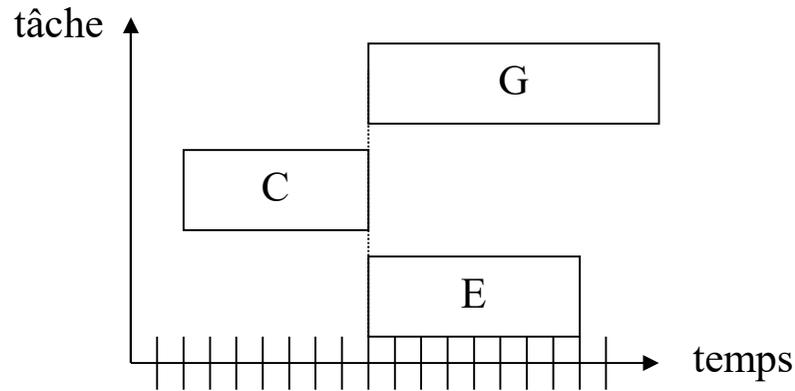
GANTT associé:



- 2 tâches **simultanées** commencent en même temps

\* Exemple :

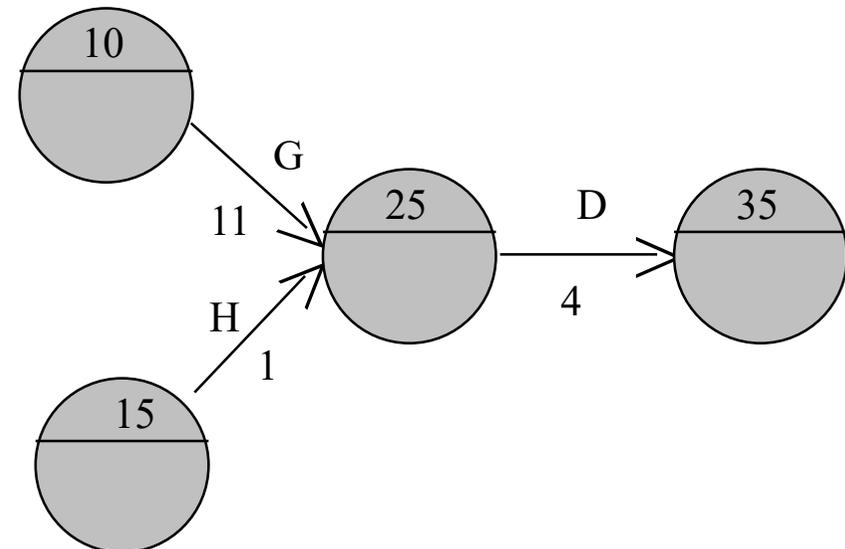
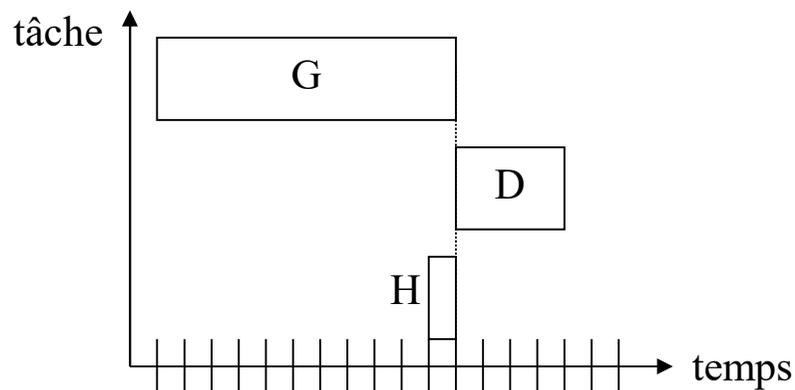
G et E ne peuvent commencer que si l'étape 10 est atteinte, donc si C est terminée



- 2 tâches **convergentes** précèdent une même tâche

\* Exemple :

D ne peut commencer que si G et H sont terminées



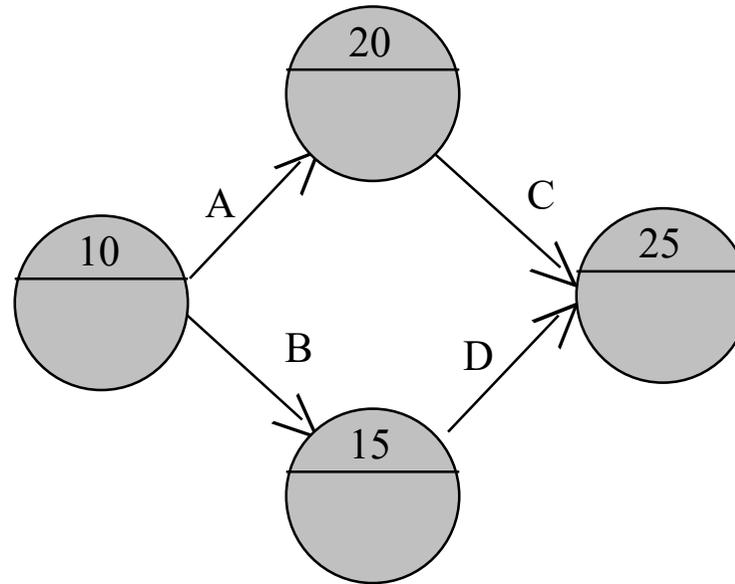
- Création d'une tâche fictive  artifice de représentation

\* Exemple :

Soit le graphe PERT suivant:

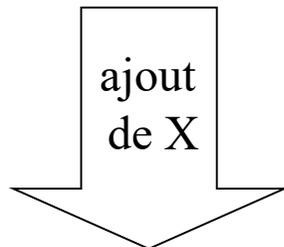
Il signifie que

- A et B sont simultanées
- C et D sont convergentes
- A précède C
- B précède D

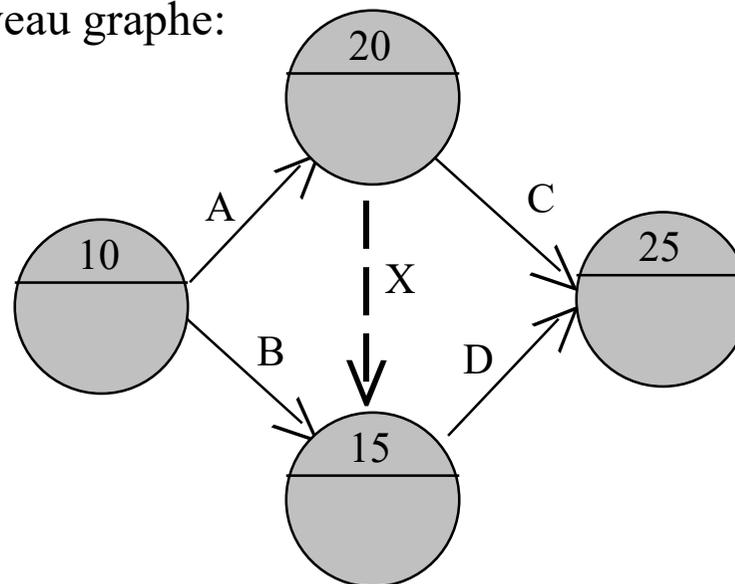


**Ajout d'une contrainte: A précède D**

=> nouveau graphe:



**X : tâche fictive de durée 0  
CE N'EST PAS UNE TACHE  
C'EST UNE CONTRAINTE**



• Exercice :

Tracer le graphe PERT représentant les conditions suivantes:

{  
A précède B  
A précède C  
D précède B

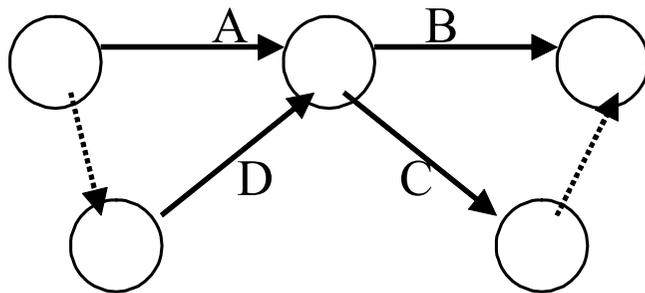
• Exercice :

Tracer le graphe PERT représentant les conditions suivantes:

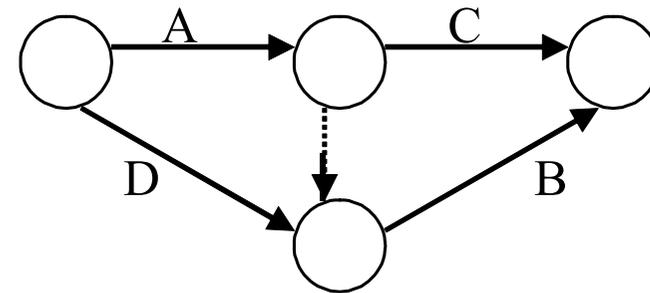
- A précède B
- A précède C
- D précède B

Solution:

1<sup>ère</sup> représentation : fausse



2<sup>o</sup> représentation : correcte



- Exemple de projet traité dans la suite:

tâches	durée (en heures)	tâches antérieures
A	4	-
B	17	-
C	20	B
D	25	A
E	8	C, D
F	7	C, D
G	3	E, F

## II.2 Construction d'un réseau PERT : méthode par niveau

### 1) Construction de la matrice des dépendances

- ⇒ matrice carré d'ordre  $n$  ( $n$  = nombre de tâches du projet).
- ⇒ chaque tâche est inscrite 2 fois : en abscisse et en ordonnée
- ⇒ se remplit ligne par ligne

est antérieure à

	A	B	C	D	E	F	G
A							
B							
C							
D							
E							
F							
G							

Soit une tâche quelconque (E par exemple).

Dans la matrice, inscrire une croix dans la case intersection entre la ligne associée à E et la colonne associée à une des tâches antérieures à E (exemple C).

	A	B	C	D	E	F	G
E			X				

Recommencer la même démarche pour E avec ses autres antécédents.

Passer à une autre ligne de la matrice.

Réitérer la procédure jusqu'à ce que toutes les lignes soient considérées.

est antérieure à

	A	B	C	D	E	F	G
A							
B							
C		X					
D	X						
E			X	X			
F			X	X			
G					X	X	

tâches	tâches antérieures
A	-
B	-
C	B
D	A
E	C, D
F	C, D
G	E, F

## 2) Niveau des tâches

Matrice des dépendances complétée par des **colonnes de niveaux**

- ⇒ **Niveau 1** = ensemble des **sommets sans antécédent**
- ⇒ Chercher les lignes sans croix et barrer ces lignes
- ⇒ Inscrire les tâches correspondantes dans la colonne du niveau 1 et sur les lignes adéquates

est antérieure à

	A	B	C	D	E	F	G	niv 1
A								A
B								B
C		X						
D	X							
E			X	X				
F			X	X				
G					X	X		

## Autres niveaux

- ⇒ Barrer les colonnes associées au niveau 1
- ⇒ Matrice réduite
- ⇒ Même démarche pour le **niveau 2**
  - ⇒ chercher les lignes vides de la matrice réduite...
  - ⇒ Jusqu'à ce que toutes les taches soient classées par niveau

est antérieure à

	A	B	C	D	E	F	G	niv 1
A								A
B								B
C		X						
D	X							
E			X	X				
F			X	X				
G					X	X		

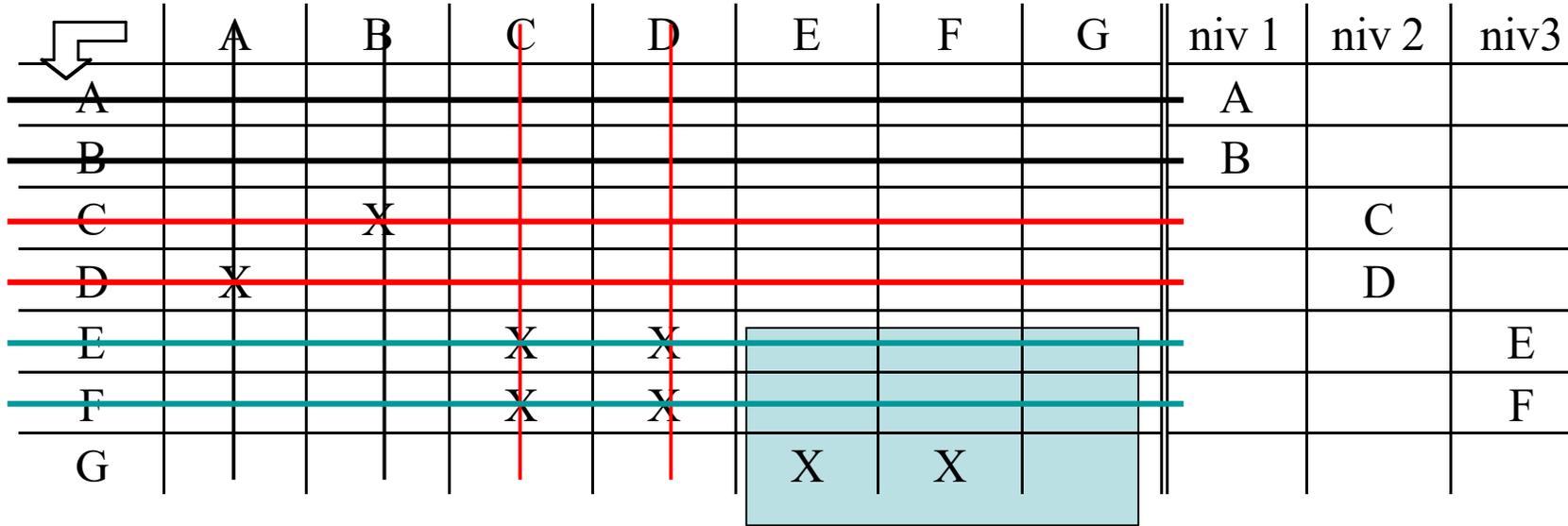
est antérieure à

	A	B	C	D	E	F	G	niv 1	niv 2
A								A	
B								B	
C		X							C
D	X								D
E			X	X					
F			X	X					
G					X	X			

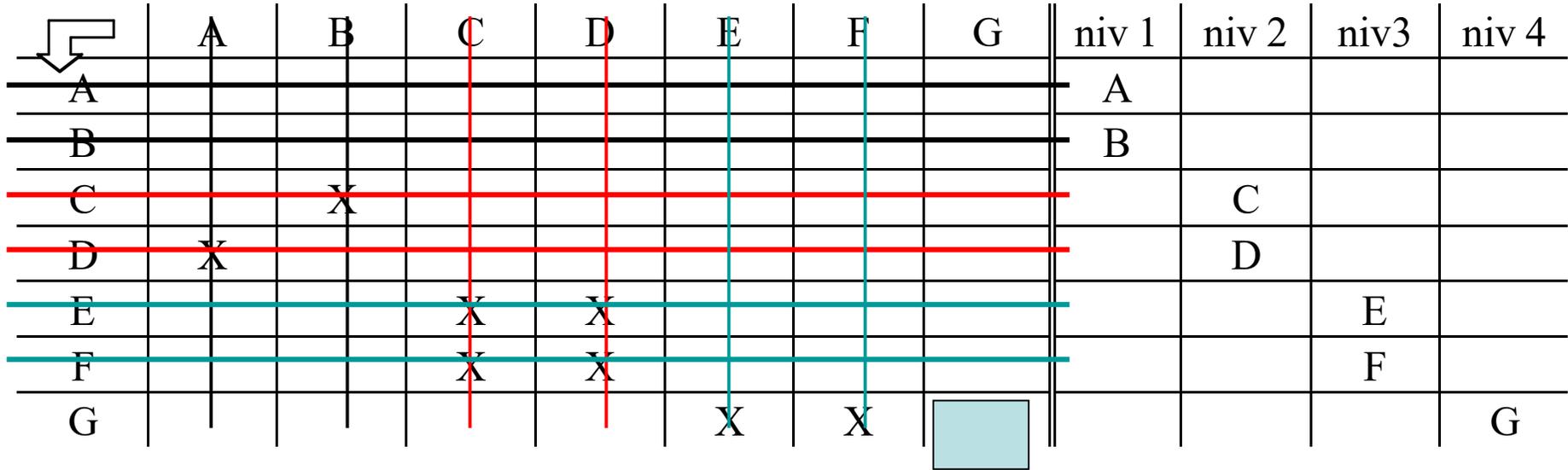
est antérieure à

	A	B	C	D	E	F	G	niv 1	niv 2
A								A	
B								B	
C		X							C
D	X								D
E			X	X					
F			X	X					
G					X	X			

est antérieure à



est antérieure à



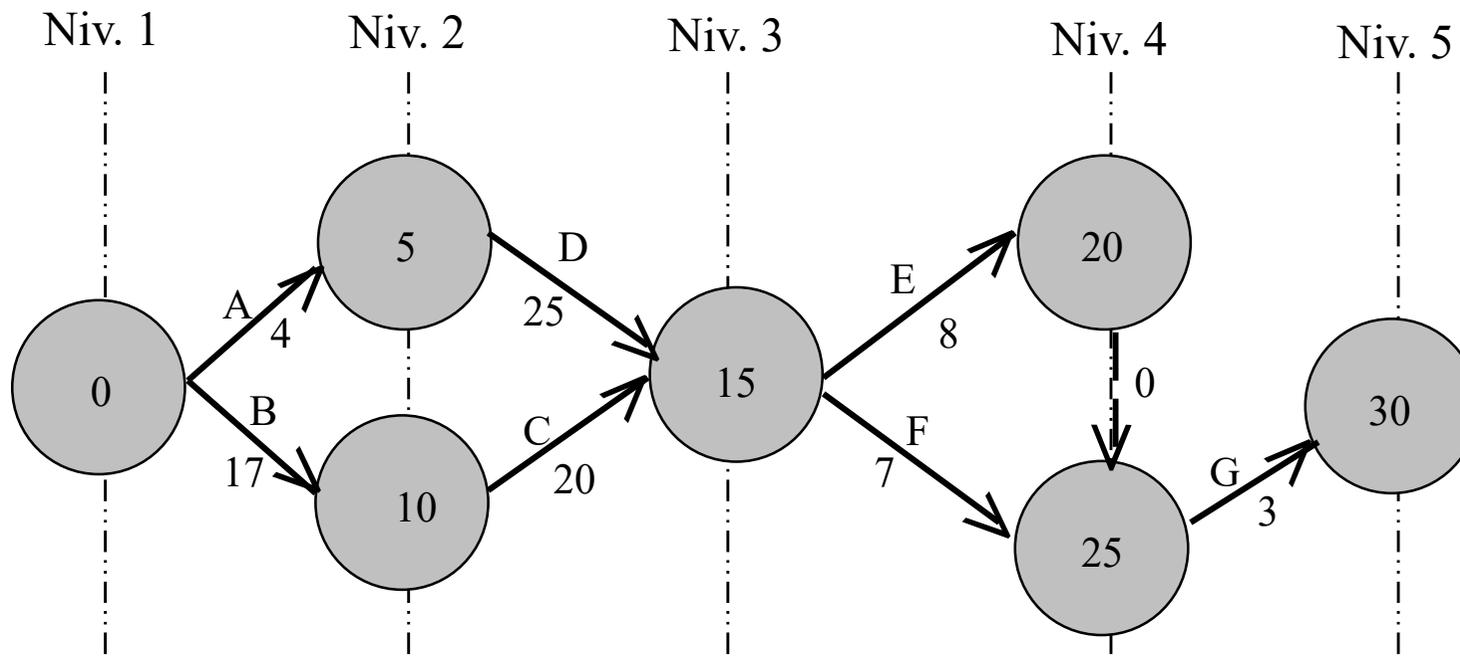
**Niveau = position des sommets des taches correspondantes**

**Niveau = position des sommets « origine » des tâches correspondantes**

Niveau 1 : A et B / Niveau 2 : C et D / Niveau 3 : E et F / Niveau 4 : G

**graphe  
PERT**

**Canevas:** colonnes $\Leftrightarrow$  niveaux  
- Taches dans colonnes  
- Étapes sur lignes de séparation



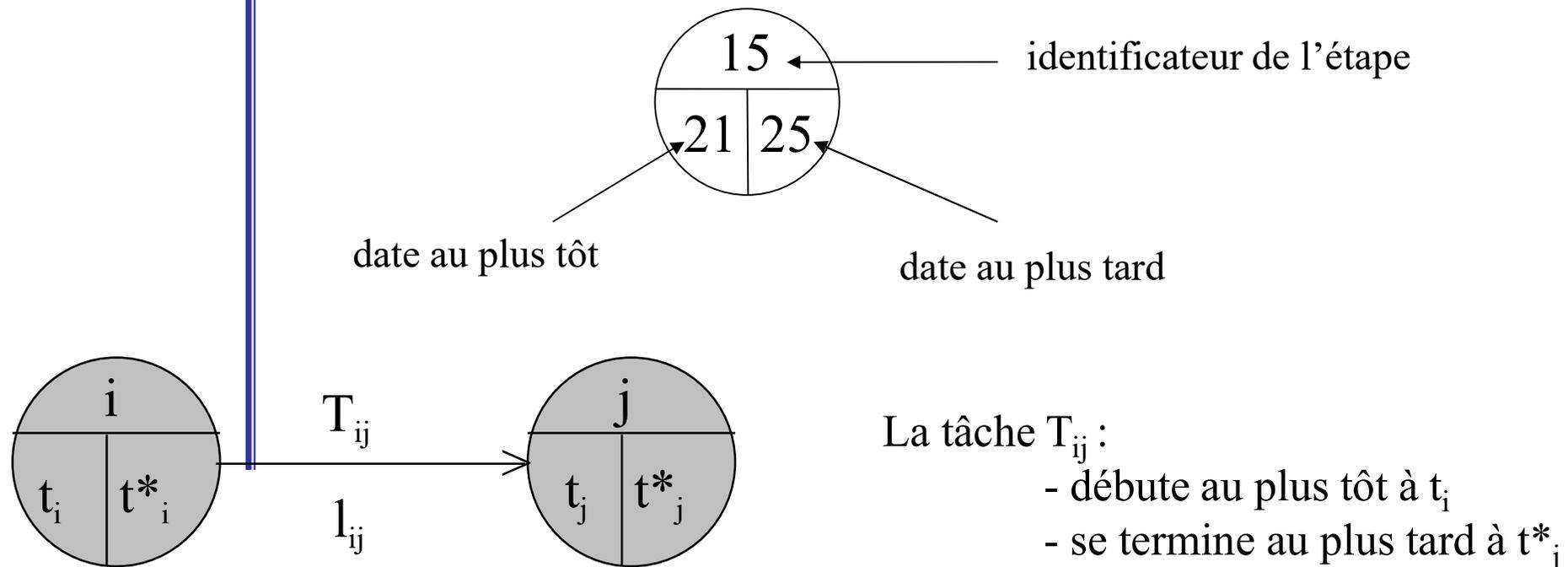
## Analyse temporelle d'un graphe PERT

**but** = déterminer : - la durée totale du projet,  
 - le degré de liberté des tâches les unes par rapport aux autres.

<= **modification du graphe** : 2 infos supplémentaires à chaque sommet :

- la date de début au plus tôt
- et la date de début au plus tard

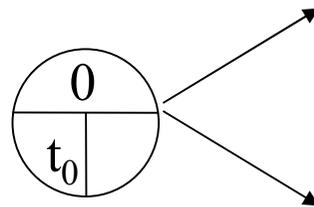
de la tâche ayant l'étape considérée comme origine.



## II.3 Calcul des dates d'exécution au plus tôt

$t_j$  date d'arrivée au plus tôt à l'étape  $E_j$

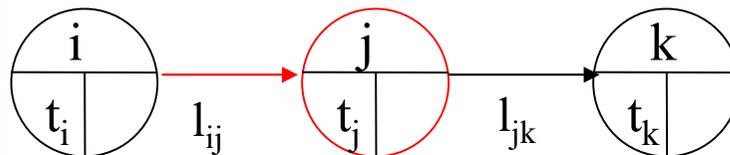
cas 1 : 1ère étape du graphe : pas de prédécesseur



$$t_0 = 0$$

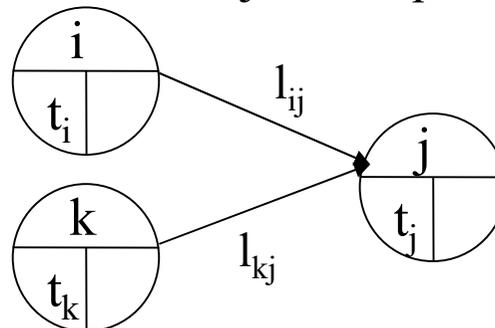
(E<sub>0</sub> est l'étape de début du projet)

cas 2 : E<sub>j</sub> est l'étape de **fin** d'exécution d'une seule tâche



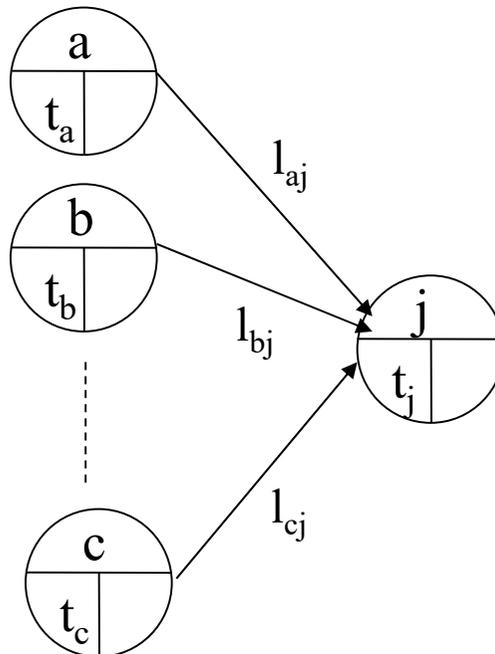
$$t_j = t_i + l_{ij}$$

cas 3 : E<sub>j</sub> est l'étape de **fin** d'exécution de 2 tâches



$$t_j = \text{Max} (t_i + l_{ij}, t_k + l_{kj})$$

cas 4 : cas général  $E_j$  est l'étape de **fin** d'exécution de plusieurs tâches



$$t_j = \text{Max} (t_i + l_{ij})$$

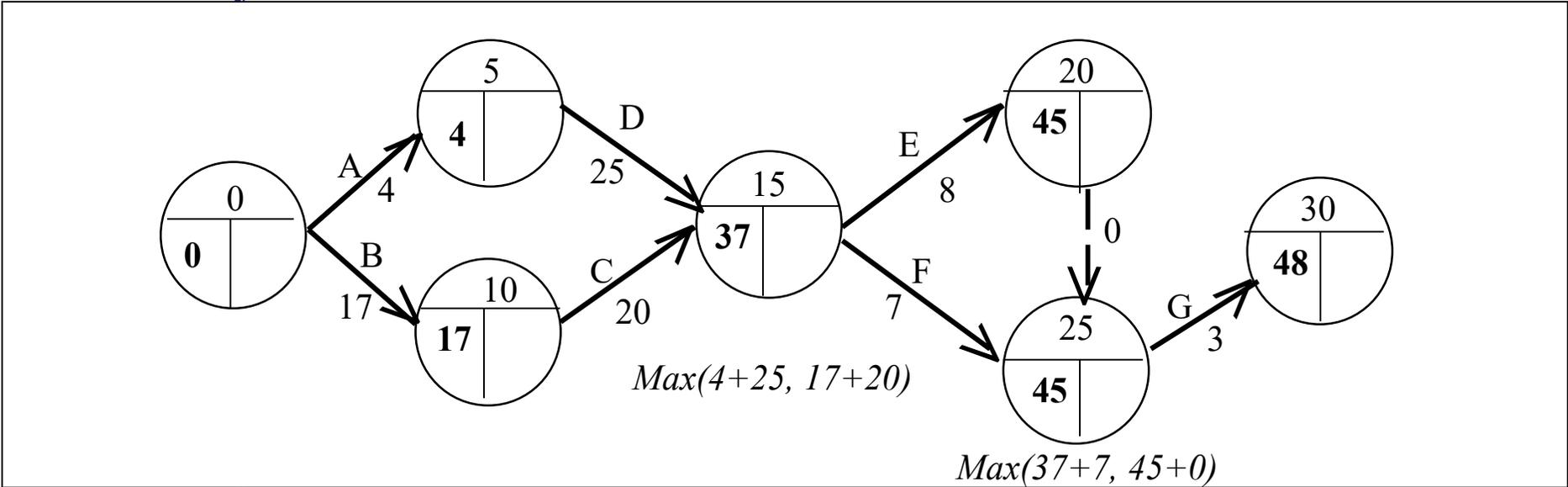
$$u=(i,j) \in U_{j^-}$$

$U_{j^-}$  : ensemble des arcs ayant l'étape  $E_j$   
pour extrémité terminale

$t_j =$  valeur du chemin de longueur maximale  
ayant  $E_0$  pour origine et  $E_j$  pour extrémité

**Parcours du graphe de gauche à droite:**  
**Additionner les durées des tâches les unes aux autres**  
**Prendre la plus grande valeur aux intersections**

**dates au plus tôt** positionner les dates dans la partie inférieure gauche des sommets

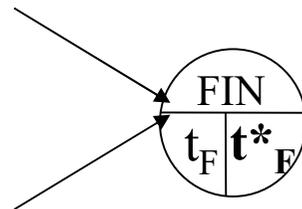


**Date au plus tôt dernière étape du PERT**  
 =  
**Durée minimale de réalisation du projet**

## II.4 Calcul des dates d'exécution au plus tard

$t_j^*$   
 $E_j$   
 date de départ au plus tard de l'étape

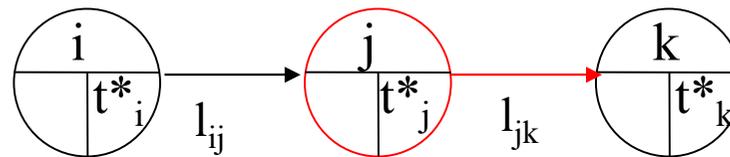
cas 1 : dernière étape du graphe  $E_F$ : pas de successeur



$$t_F^* = t_F$$

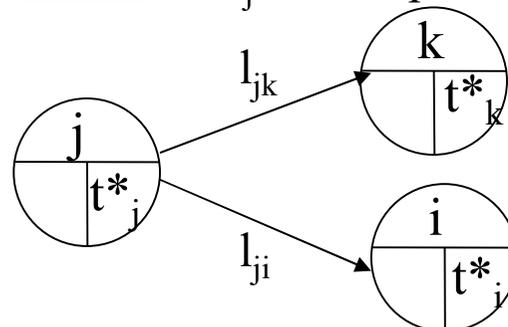
(pour réaliser la durée minimale)

cas 2 :  $E_j$  est l'étape de **début** d'exécution d'une seule tâche



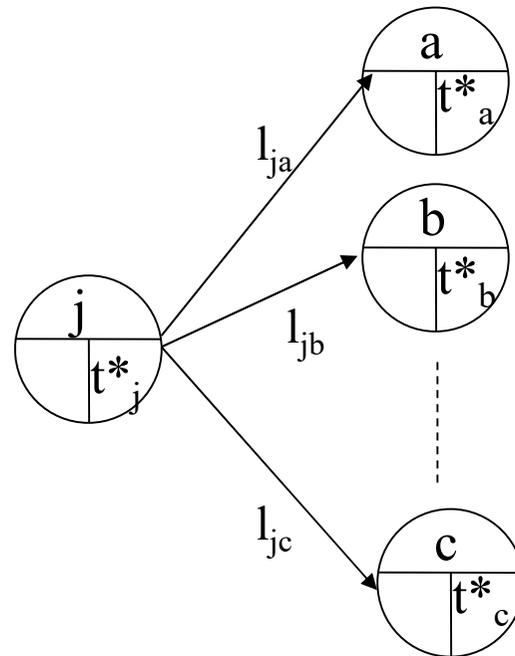
$$t_j^* = t_k^* - l_{jk}$$

cas 3 :  $E_j$  est l'étape de **début** d'exécution de 2 tâches



$$t_j^* = \text{Min} (t_i^* - l_{ji}, t_k^* - l_{jk})$$

cas 4 : cas général  $E_j$  est l'étape de **début** d'exécution de plusieurs tâches

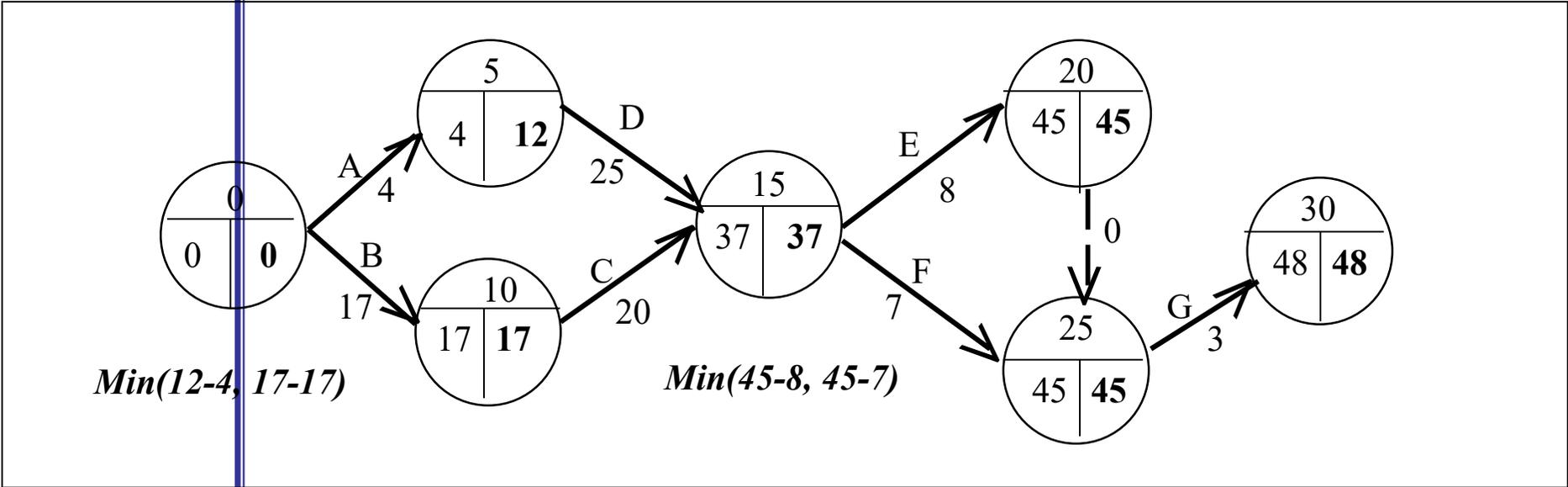


$$t^*_j = \text{Min} (t^*_i - l_{ji})$$
$$u=(j,i) \in U_{j+}$$

$U_{j+}$  : ensemble des arcs ayant l'étape  $E_j$   
pour extrémité initiale

**Parcours du graphe de droite   gauche :**  
**Soustraire les dur es des t ches les unes aux autres**  
**Prendre la plus petite valeur aux intersections**

**dates au plus tard** positionner les dates dans la partie inf rieure droite des sommets



**Date au plus tard premi re  tape du PERT = 0**

## II.5 Détermination des marges

### Marges

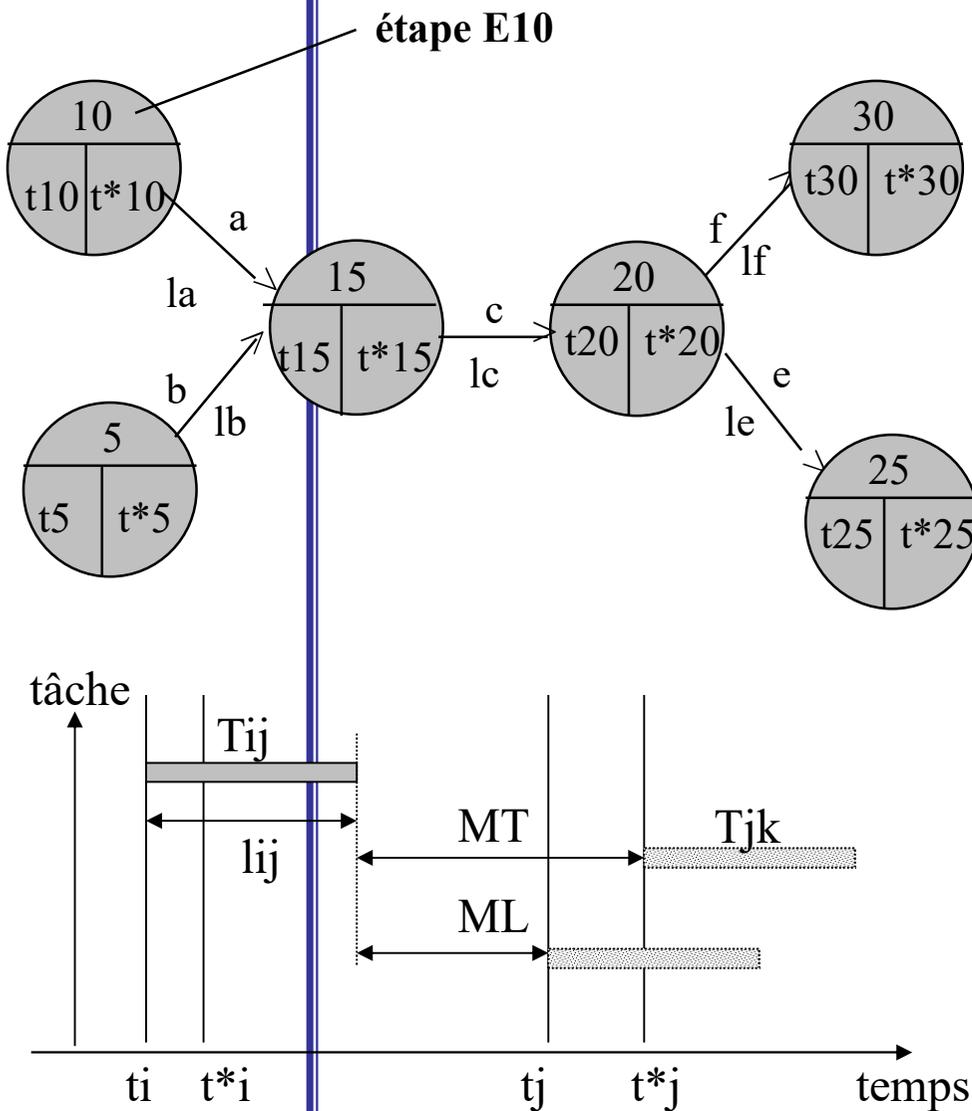
- degré de liberté dans l'accomplissement des tâches

- utiles pour : \* répartir les moyens

\* réduire les temps

\* jouer sur les coûts

## II.5 Détermination des marges



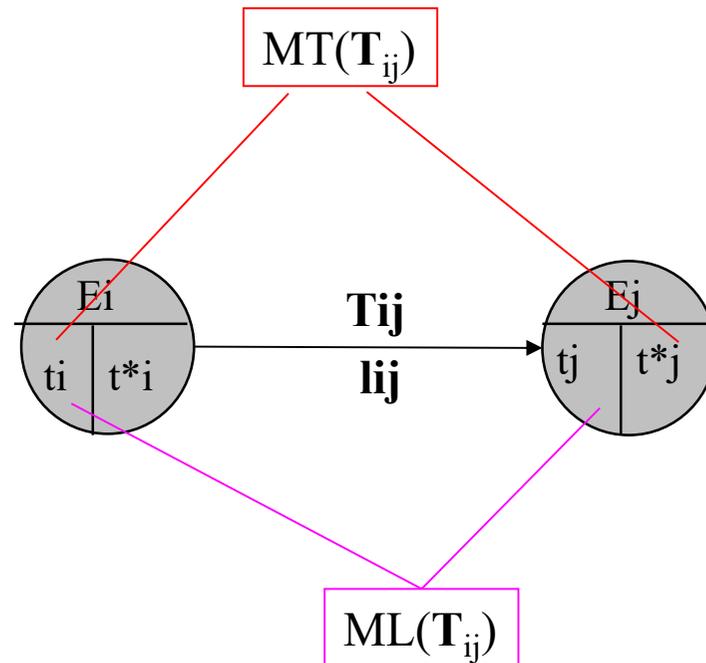
- marge **totale** de  $T_{ij}$ : retard toléré sans modification de la durée totale mais retarde les successeurs

$$MT(T_{ij}) = t^*_j - t_i - l_{ij}$$

- marge **libre** de  $T_{ij}$ : retard toléré sans modification pour la suite du projet

$$ML(T_{ij}) = t_j - t_i - l_{ij}$$

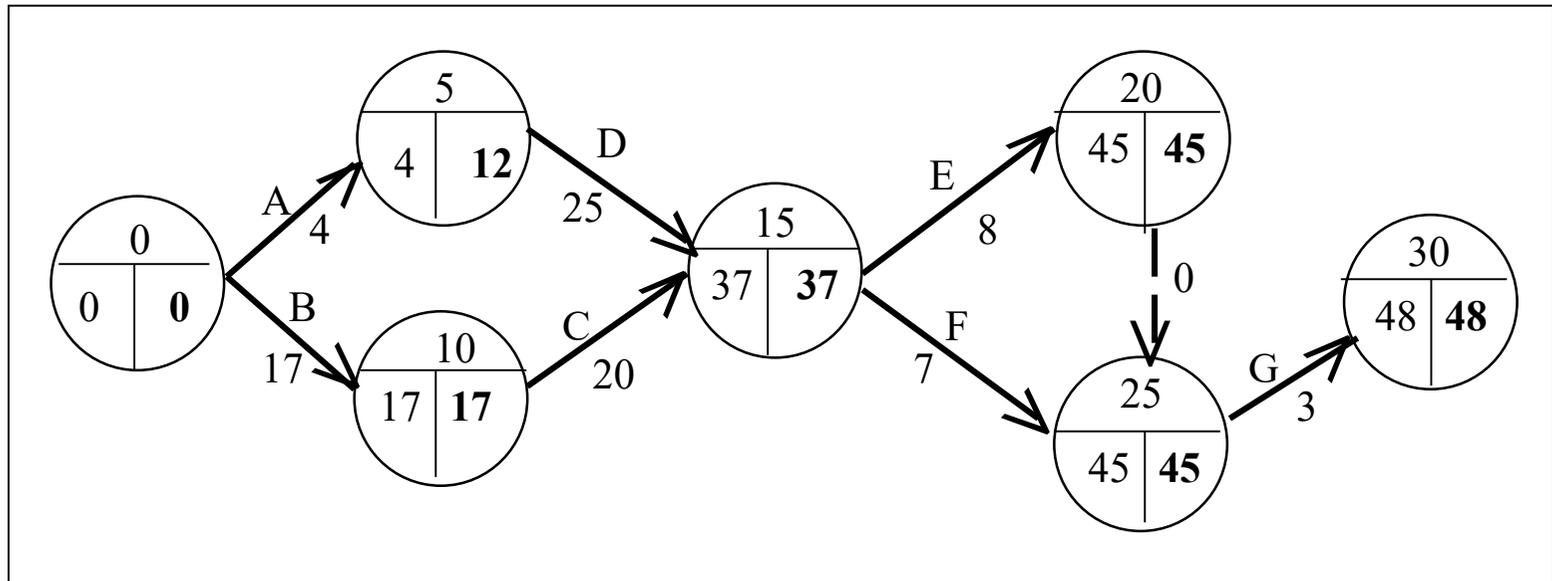
## II.5 Détermination des marges



$$MT(T_{ij}) = t^*_j - t_i - l_{ij}$$

$$ML(T_{ij}) = t_j - t_i - l_{ij}$$

## II.5 Détermination des marges



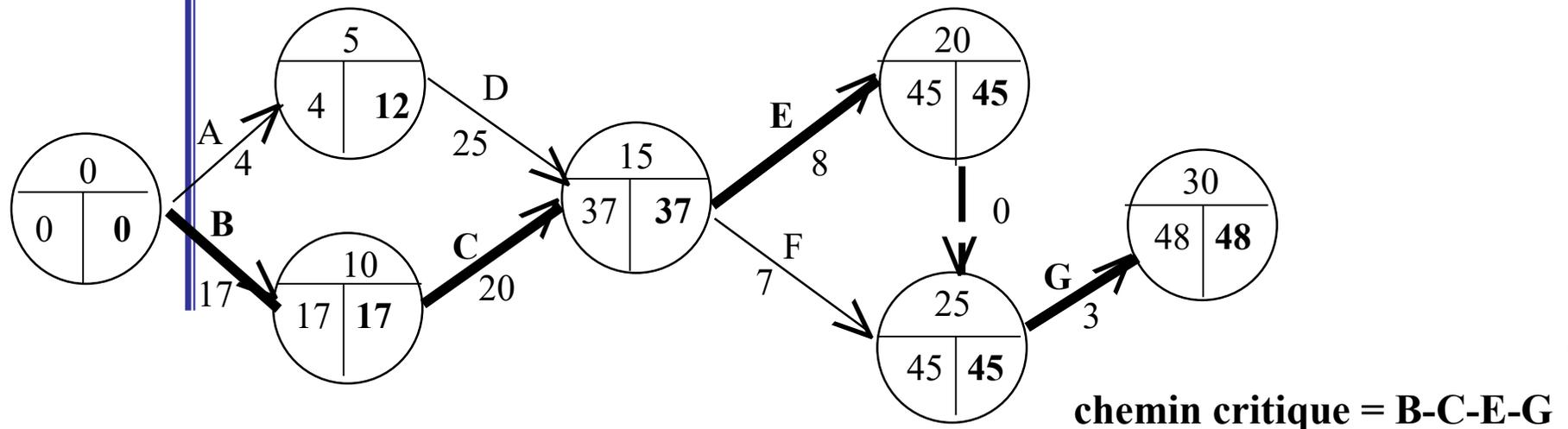
tâches	MT	$\geq$	ML
A	$8 = 12 - 0 - 4$		$0 = 4 - 0 - 4$
B	$0 = 17 - 0 - 17$		0
C	$0 = 37 - 17 - 20$		0
D	$8 = 37 - 4 - 25$		$8 = 37 - 4 - 25$
E	0		0
F	$1 = 45 - 37 - 7$		1
G	0		0

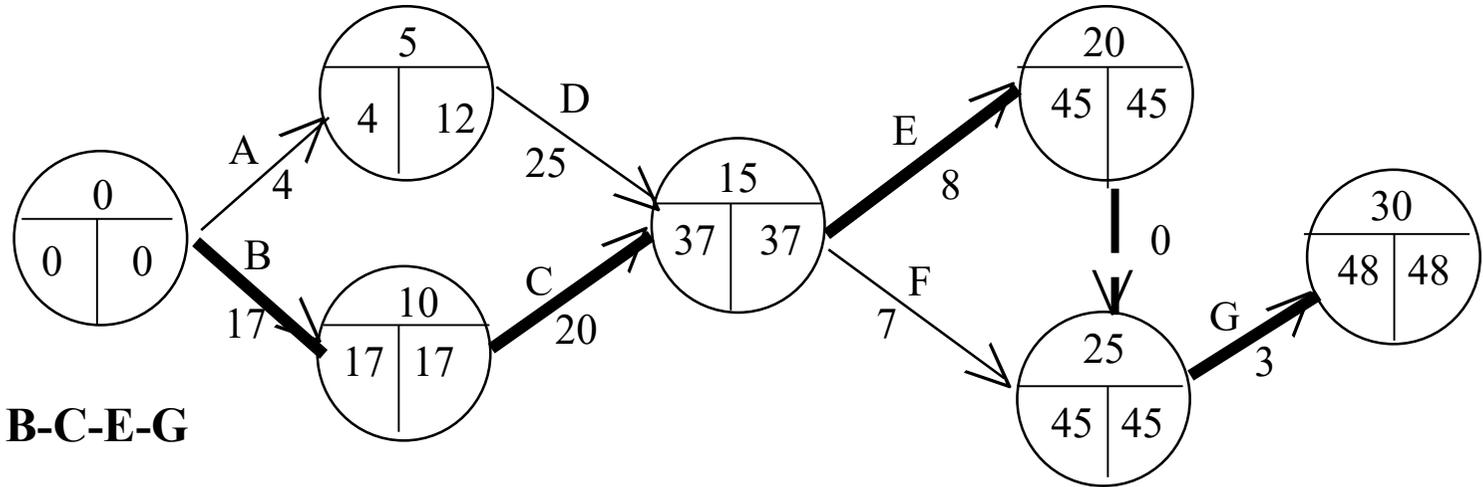
## II.6 Détermination du chemin critique

**chemin critique** = chemin plus long que tous les autres  
constitué de tâches critiques et d'étapes critiques

**tâches critiques** : la tâche  $T_{ij}$  est critique  
ssi **marge totale( $T_{ij}$ ) = 0**  $\longrightarrow$   $\left\{ \begin{array}{l} \text{date au plus tôt } t_i = \text{date au plus tard } t^*i \\ \text{date au plus tôt } t_j = \text{date au plus tard } t^*j \end{array} \right.$

tâches du chemin critique = fixent le délai de réalisation du projet  
**délai minimal** = somme des durées des tâches du chemin critique

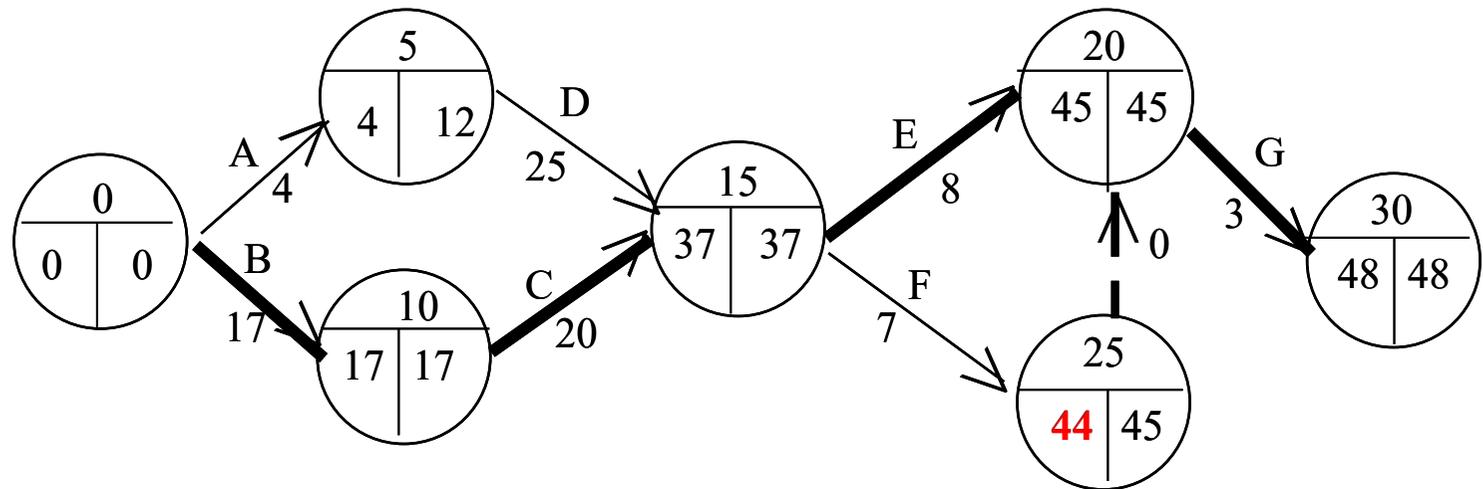




chemin critique = B-C-E-G

Attention: F non critique:  $MT(F) \neq 0$

Autre construction équivalente ⇕



Réseau PERT du projet Mididanse

