

Algorithmique des graphes

Projet PERT

Prépa INP

Septembre 2023

1 Introduction

PERT (Program Evaluation and Review Technique)¹ est une méthode de gestion de projet. Il permet notamment d'organiser les tâches d'un projet en fonction de leurs *durées* respectives et de leurs *dépendances*. En fonction de ces durées et de ces dépendances, le PERT prévoit une *date de début* et une *date de fin* pour chaque tâche. Aussi, le PERT identifie les tâches critiques, c'est-à-dire les tâches qui ne doivent souffrir d'aucun retard sous peine de retarder l'ensemble du projet.

Dans le cadre de cette étude, on s'intéressera aux systèmes de tâches où une tâche est représentée uniquement par une durée exacte et les liaisons de précédence avec les autres tâches. Une tâche ne peut commencer que si toutes les tâches qui la précèdent se sont terminées. A titre d'exemple, une description approximative des tâches pour la construction d'une maison individuelle est donnée dans l'annexe A.

Le PERT est aussi utilisé pour suivre le déroulement d'un projet une fois commencé. Un *compte rendu d'exécution* recense donc, à une date donnée, pour chaque tâche son état d'avancement. On peut ainsi reporter les avances et les retards de chaque tâche et, par suite, (re)prévoir, si besoin est, les dates au plus tôt et au plus tard de chacune des tâches. Par suite, il sera possible de communiquer, après chaque compte rendu d'exécution, sur l'avance ou le retard éventuels et donner de nouvelles dates de terminaison possibles de fin du projet.

Enfin, une fois le projet terminé, il est essentiel de garder une trace du déroulement du projet. Il sera donc nécessaire de prévoir un dossier du projet, où un chapitre historique permettra de reconstituer le déroulement du projet.

1.1 Modèle réduit

Pour illustrer le travail demandé, nous considérons la description de tâches illustrée par le tableau suivant (pour simplifier, seuls 3 suivis S0,S1,S2 sont ici prévus) :

Identificateur	Description	Durée	Précédente(s)	S0	S1	S2
D	Tâche de début	2 mois				
I1	Tâche intermédiaire 1	1 mois	D			
I2	Tâche intermédiaire 2	3 mois	D			
I3	Tâche intermédiaire 3	2 mois	D			
F	Tâche de fin	2 mois	I1 I2 I3			

1. <https://fr.wikipedia.org/wiki/PERT>

Dans ce système de tâches, il faudra d'abord mener la tâche D, puis les tâches I1, I2 et I3 pourront être menées en parallèle. Enfin la tâche F sera menée.

La tâche D commencera à T0.

La tâche I1 commencera au plus tôt à T0+2, se terminera au plus tôt à T0+3 et devra se terminer au plus tard à T0+5..

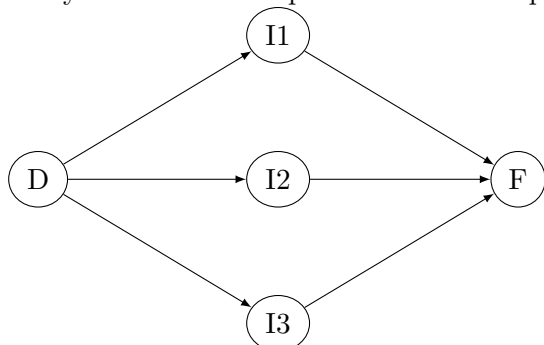
La tâche I2 commencera au plus tôt à T0+2, se terminera au plus tôt à T0+5 et devra se terminer au plus tard à T0+5.

La tâche I3 commencera au plus tôt à T0+2, se terminera au plus tôt à T0+4 et devra se terminer au plus tard à T0+5.

La tâche F commencera au plus tôt à T0+5, se terminera au plus tôt et au plus tard à T0 + 7.

La tâche I2 est une tâche critique, en effet son retard entrainera un retard global du projet.

Ce système de tâches peut être visualisé par la figure suivante :



NB. A ce stade, on raisonne par rapport à une date initiale et des *durées*. On ne prend pas en compte des dates effectives qui intègreraient les WE, les jours fériés, ...

2 Travail demandé

1. L'interface du projet sera définie à partir du format standard csv. Dans une première phase du projet, il sera décidé une définition commune (à toute la promotion) des fichiers d'interface nécessaires à chaque phase du projet. Les fichiers d'interface comportent les données en entrée (tâches, durées, précédences, exécution) et les données en sortie (tâche, dates, chemin critique).
2. Une documentation utilisateur comportant un mode d'emploi du logiciel (pour un **non programmeur**).
3. Un rapport projet comportant : les résultats produits pour un jeu d'entrée donné seront :
 - Les précédences de tâches seront explicitées par la visualisation d'un graphe de tâches.
 - analyse de faisabilité de l'ordonnancement des tâches.
 - Les chemins critiques seront explicités sur la visualisation du graphe de tâches.
 - les comptes rendu d'exécution.
 - l'historique du projet.
4. Un rapport de documentation de la programmation du projet comportant :
 - des jeux de tests pertinents de systèmes de tâches à l'aide des logiciels de test vus en TP.
 - Le code source Python documenté. (Ce code pourrait comporter une modélisation UML)

A Description approximative des tâches pour la construction d'une maison individuelle

A.1 Description des tâches

Identificateur	Description	Durée	Précédente(s)	S0	S1	S2
PC	Permis de construire	2 mois				
F	Fondations	1 semaine	P			
GE1	Passage des gaines et évacuations	3 jours	F			
DRC	Dalle rez de chaussée	1 semaine	GE1			
MP	Murs porteurs	2 semaines	DRC			
DP	Dalles plafond	1 semaine	DP			
GE2	Passage gaines et évacuation	3 jours	DP			
T	Toiture	2 semaines	GE2			
F	Fenêtres	1 semaine	T			
PE	Portes extérieures	3 jours	T C			
IE	Installation électrique et évacuation	3 jours	GE2			
C	Chape	1 semaine	DP			
C1	Carrelage du sol	1 semaine	C			
P	Parquets	1 semaine	C			
CM	Cloisons et menuiserie intérieure	10 jours	F C1 P FC			
FC	Finition des cloisons	1 semaine	CM			
IC	Implantation de la cuisine	1 semaine	FC			
IW	Implantation des wc	3 jours	FC			
IS	Implantation des salles de bain	1 semaine	FC			
PP	Peinture des plafonds	1 semaine	IC IW IS			
PM	Peinture des murs	1 semaine	PP			
S	Serrurerie	3 jours	P CM			
R1	Revêtement des sols (moquettes)	2 jours	C S			
R	Réception de la maison	0.5 jour	PM S			

B Le format CSV

Le format CSV² (Comma-separated values) est un format associé à des tableaux. Chaque rangée (ligne) de ce tableau est représentée par une chaîne de caractères. Les rangées sont séparées par un caractère de fin de ligne. Les colonnes d'une ligne sont séparées par des ,.

Il est utilisé comme format de sauvegarde et d'échange par les outils de bureautique (suite OpenOffice³, word-excel⁴) ou encore d'analyse statistique (⁵). Par suite, il est possible de réutiliser ces outils, en utilisant ce format.

A titre d'exemple, soit le tableau suivant :

2. https://fr.wikipedia.org/wiki/Comma-separated_values

3. <https://www.openoffice.org>

4. <https://www.microsoft.com/fr-fr/microsoft-365/free-office-online-for-the-web>

5. <https://pandas.pydata.org>

Identificateur	Description	Durée	Précédente(s)	S0	S1	S2
D	Tâche de début	2 mois				
I1	Tâche intermédiaire 1	1 mois	D			
I2	Tâche intermédiaire 2	3 mois	D			
I3	Tâche intermédiaire 3	2 mois	D			
F	Tâche de fin	2 mois	I1 I2 I3			

Le tableau précédent sera représenté au format CSV ainsi :

```
"Identificateur","Description","Durée","Précédente(s)","S0","S1","S2"
"D","Tâche de début","2 mois"
"I1","Tâche intermédiaire 1","1 mois","D"
"I2","Tâche intermédiaire 2","3 mois","D"
"I3","Tâche intermédiaire 3","2 mois","D"
"F","Tâche de fin","2 mois","I1 I2 I3"
```

B.1 Ecriture d'un fichier CSV

```
# -- coding: utf-8 --
import csv

def to_csv(name_csv_file: str, fieldnames: list[str], rows: list[list[str]]) -> None:
    """
    Args:
        name_csv_file (str): nom du fichier csv (sans suffixe)
        fieldnames (list[str]): liste des noms de champs
        rows (list[list[str]]): liste des lignes (liste de colonnes)
    """
    with open(name_csv_file+'.csv','w', encoding="utf-8") as csvfile:
        csvwriter = csv.writer(csvfile, delimiter=',', quoting=csv.QUOTE_ALL)
        csvwriter.writerow(fieldnames)
        for r in rows:
            csvwriter.writerow(r)
```

exemple :

```
to_csv("S",[ "Identificateur", "Description", "Durée", "Précédente(s)", "S0", "S1", "S2" ]
,[[ "D", "Tâche_de_début", "2_mois" ]
, [ "I1", "Tâche_intermédiaire_1", "1_mois", "D" ]
, [ "I2", "Tâche_intermédiaire_2", "3_mois", "D" ]
, [ "I3", "Tâche_intermédiaire_3", "2_mois", "D" ]
, [ "F", "Tâche_de_fin", "2_mois", "I1_I2_I3" ]
])
```

L'exécution de ce programme crée le fichier 'S.csv' dont le contenu est donné dans la section précédente.

B.2 Lecture d'un fichier CSV

```
# -- coding: utf-8 --
import csv

def from_csv(name_csv_file: str) -> list[list[str]] :
    with open(name_csv_file+'.csv','r', encoding="utf-8") as csvfile:
        csvreader = csv.reader(csvfile, delimiter=',', quoting=csv.QUOTE_ALL)
        l = []
        for r in csvreader:
            l.append(r)
        return l
```

exemple :

```
print(from_csv("S"))
```

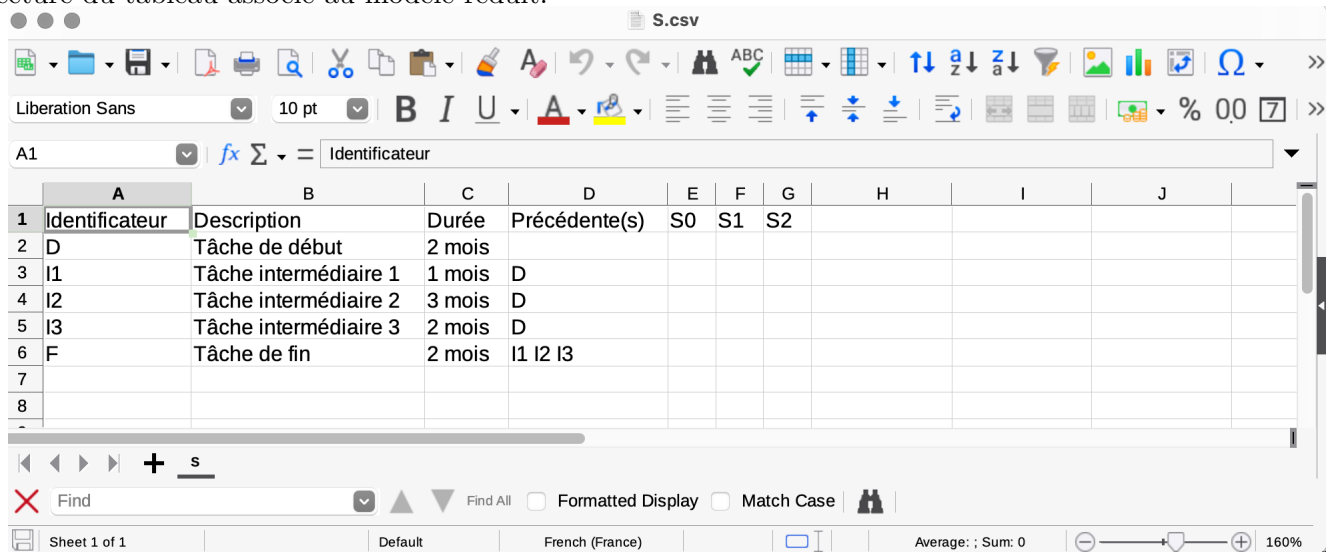
L'exécution de ce programme affiche le texte suivant (sur une ligne) :

```
[['Identificateur', 'Description', 'Durée', 'Précédente(s)', 'S0', 'S1', 'S2'],
 ['D', 'Tâche de début', '2 mois'], ['I1', 'Tâche intermédiaire 1', '1 mois', 'D'],
```

```
[ 'I2', 'Tâche intermédiaire 2', '3 mois', 'D'],
[ 'I3', 'Tâche intermédiaire 3', '2 mois', 'D'],
[ 'F', 'Tâche de fin', '2 mois', 'I1 I2 I3']]
```

B.3 Visualisation d'un fichier CSV

Un fichier CSV peut être relu par le logiciel LibreOffice. La copie d'écran suivante illustre la relecture du tableau associé au modèle réduit.



	A	B	C	D	E	F	G	H	I	J
1	Identificateur	Description	Durée	Précédente(s)	S0	S1	S2			
2	D	Tâche de début	2 mois							
3	I1	Tâche intermédiaire 1	1 mois	D						
4	I2	Tâche intermédiaire 2	3 mois	D						
5	I3	Tâche intermédiaire 3	2 mois	D						
6	F	Tâche de fin	2 mois	I1 I2 I3						
7										
8										