Kenza BOUZERNA - 1807/BINFO2/3_MathAppliInfo/23
MATHEMATIQUES APPLIQUEES
EXAMEN 17 JANVIER 2024

TABLE DES MATIÈRES

RAPPELONS L'ÉNONCÉ

Afin d'arrondir vos fins de mois vous avez accepté de travailler le soir chez Pizza-Domicile. Pour les dernières livraisons de la soirée, le patron vous demande de passer aux adresses numérotées de 2 à 11 ci-dessous. Comme vous aimeriez rentrer tôt pour bûcher votre contrôle du lendemain, votre collègue accepte de prendre à sa charge un certain nombre de livraisons. Il vous demande cependant de rallier le point de fabrication (1) à votre domicile (12) en prenant **le chemin le plus long**. Il prendra alors à sa charge les points par lesquels vous ne serez pas passé. Vous établissez donc rapidement un plan des liaisons possibles entre les différentes adresses et les durées de trajet approximatives :

DÉPART / ARRIVÉE	1	2	3	4	5	6	7	8	9	10	11	12
1	∞	10	∞	07	02	∞	∞	∞	∞	∞	∞	∞
2	∞	∞	12	∞	08	∞	∞	∞	∞	∞	∞	∞
3	∞	∞	∞	∞	21	∞	∞	∞	∞	∞	∞	∞
4	∞	∞	∞	∞	∞	∞	13	∞	∞	∞	∞	∞
5	∞	∞	∞	02	∞	∞	15	∞	∞	∞	∞	∞
6	∞	∞	∞	∞	04	∞	12	∞	∞	∞	∞	∞
7	∞	∞	∞	∞	∞	∞	∞	10	08	∞	∞	∞
8	∞	∞	∞	∞	∞	∞	∞	∞	03	04	∞	05
9	∞	∞	∞	∞	∞	∞	∞	∞	∞	07	∞	28
10	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	15	09
11	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	04
12	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞

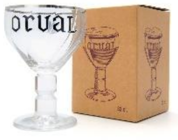
Ce tableau indique par exemple qu'il est possible de rallier le point 4 au point 7 en 13 minutes. Déterminez le trajet que vous allez emprunter et le temps qu'il vous faudra pour rallier votre domicile.

Afin de s'implanter sur le marché, l'entreprise Pizza-Domicile fait la vente sur place de 3 types de pizzas à prix très bas, la Royale, la Margarita, et la Spéciale. Cependant cette offre n'est proposée que dans la mesure des stocks d'ingrédients disponibles. Le tableau ci-dessous dresse la liste des quantités d'ingrédients (en grammes) que doivent contenir ces 3 pizzas :

INGRÉDIENTS (EN G)	GRUYÈRE	JAMBON (Halal)	CHAMPIGNONS	LARDONS (Halal)
ROYALE	40	50	20	0
MARGARITA	20	0	40	30
SPÉCIALE	50	60	20	40

D'autre part, afin de limiter la demande le patron ne veut pas, pour ces promotions, utiliser plus de 3000 g de gruyère, 3100 g de jambon, 2000 g de champignons et 1000 g de lardons. Les bénéfices réalisés sur ces pizzas sont de 1 €, 2 € et 2 € respectivement sur la Royale, la Margarita et la Spéciale.

Le patron ayant appris que vous effectuiez des études d'informatique vous demande de déterminer quelles sont les quantités de pizzas de chaque type qu'il doit fabriquer pour maximiser son bénéfice sur cette promotion.



Vu le succès commercial de cette entreprise, le patron envisage de créer une succursale à la Chambre d'Amour. Cependant, il est indispensable qu'il puisse ouvrir avant les congés d'été afin de minimiser rapidement les frais occasionnés par les emprunts nécessaires. Pour cela il fait encore appel à vos services et vous dresse la liste de toutes les tâches qui devront être effectuées avant l'ouverture. Celles-ci sont données dans le tableau suivant ainsi que les tâches précédentes et la durée en jour de chacune de ces tâches

x	A	B	C	D	E	F	G	H	I	J	K	L
P(x)	-	-	A	A	B	A,D	C	A,C,D	B	E,F,I	C,G,H	F,J
Durée	15	12	25	5	13	7	31	14	8	2	30	7

Déterminez les calendriers au plus tôt, au plus tard, les tâches critiques, la durée du projet, les marges totales et libres.

PROBLÈME 1

Pour résoudre ce problème, on peut utiliser l'algorithme de Dijkstra. L'algorithme de Dijkstra est un algorithme de recherche de plus court chemin utilisé en informatique et en théorie des graphes. Cet algorithme permet de trouver le chemin le plus court entre un nœud source et tous les autres nœuds dans un graphe pondéré, c'est-à-dire un graphe où chaque arête a un poids ou une valeur associée.

Voici les étapes de l'algorithme de Dijkstra :

1. Initialisation : Assignez une valeur de distance infinie à tous les nœuds, sauf au nœud source pour lequel la distance est de 0. Créez un ensemble de nœuds non traités.
2. Sélection du nœud le plus proche : À chaque étape, sélectionnez le nœud non traité avec la distance minimale parmi ceux encore non traités. Pour le premier tour, le nœud source aura une distance de 0.
3. Mise à jour des distances : Pour chaque nœud adjacent au nœud sélectionné, mettez à jour la distance totale depuis le nœud source si le chemin actuel est plus court que la distance précédemment connue. Cela signifie que si la distance du nœud source au nœud sélectionné, plus la distance de l'arête entre le nœud sélectionné et le nœud adjacent, est inférieure à la distance actuellement connue du nœud source au nœud adjacent, mettez à jour la distance.
4. Marquer le nœud sélectionné : Une fois que toutes les distances adjacentes ont été mises à jour, marquez le nœud actuellement sélectionné comme traité. Cela signifie que la distance la plus courte depuis le nœud source jusqu'à ce nœud a été trouvée.
5. Répéter : Répétez les étapes 2 à 4 jusqu'à ce que tous les nœuds aient été traités.

À la fin de l'algorithme, les distances minimales depuis le nœud source jusqu'à tous les autres nœuds du graphe sont connues. De plus, l'algorithme maintient un ensemble de prédécesseurs, ce qui permet de reconstruire les chemins les plus courts.

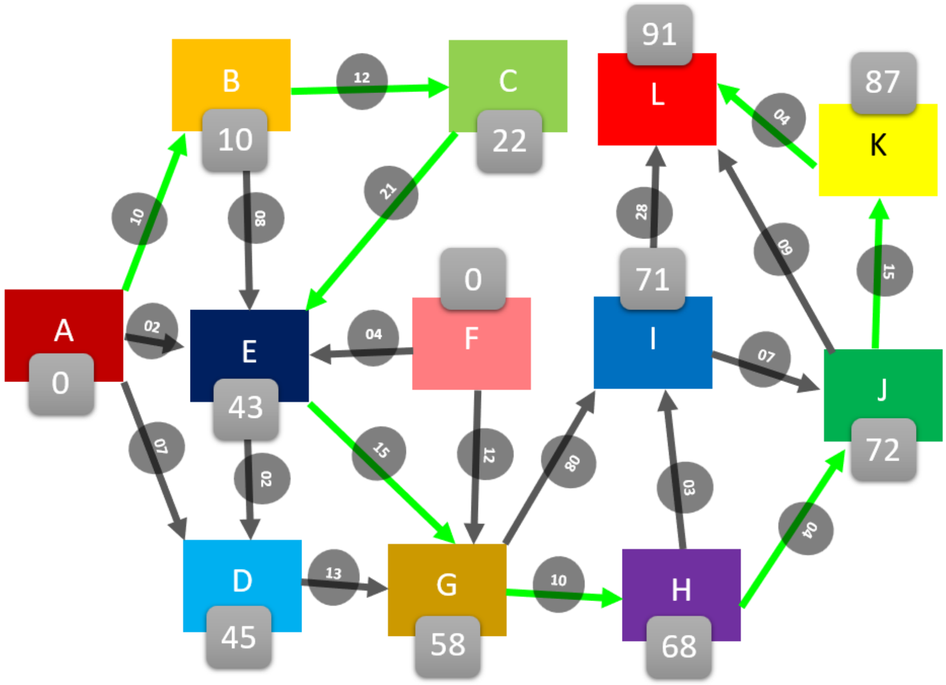
Dans notre cas, nous chercherons non pas le chemin le plus court mais le chemin le plus long. A noter que le sommet source est le sommet 1 et le sommet destination est le sommet 12.

Cependant, pour des raisons de clarté, nous allons renommer les sommets non pas par des nombres mais par des lettres comme suit :

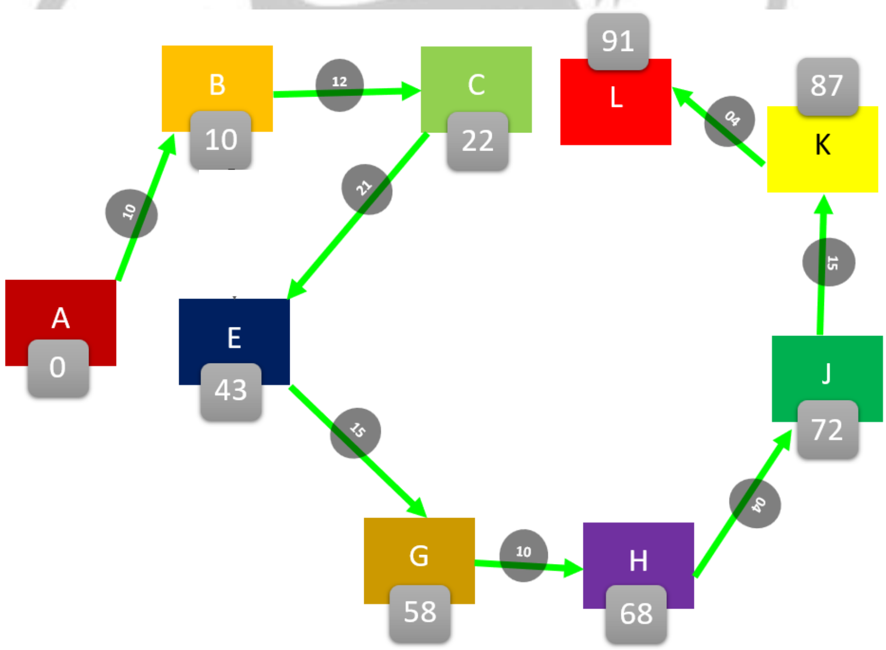
DÉPART / ARRIVÉE	A	B	C	D	E	F	G	H	I	J	K	L
A	∞	10	∞	07	02	∞	∞	∞	∞	∞	∞	∞
B	∞	∞	12	∞	08	∞	∞	∞	∞	∞	∞	∞
C	∞	∞	∞	∞	21	∞	∞	∞	∞	∞	∞	∞
D	∞	∞	∞	∞	∞	∞	13	∞	∞	∞	∞	∞
E	∞	∞	∞	02	∞	∞	15	∞	∞	∞	∞	∞
F	∞	∞	∞	∞	04	∞	12	∞	∞	∞	∞	∞
G	∞	∞	∞	∞	∞	∞	∞	10	08	∞	∞	∞
H	∞	∞	∞	∞	∞	∞	∞	∞	03	04	∞	05
I	∞	∞	∞	∞	∞	∞	∞	∞	∞	07	∞	28
J	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	15	09
K	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	04
L	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞

La confection du graphe se fera avec l’outil Word (oui, Word est un très bon outil !).

L'algorithme de Dijkstra s'exécute comme suit :



Si nous tenons compte uniquement du chemin le plus long, nous obtenons ceci :



Ce tableau indique que le chemin le plus long entre les sommets A et L est le suivant :

`A -> B -> C -> E -> G -> H -> J -> K -> L`

La durée de ce chemin est donc de 0 + 10 + 12 + 21 + 15 + 10 + 04 + 15 + 04 = 91 minutes.

PROBLÈME 2

Pour résoudre ce problème, on peut utiliser la programmation linéaire.

Le programme linéaire est le suivant :

VARIABLES PIZZA

- x1 = nombre de Royale
- x2 = nombre de Margarita
- x3 = nombre de Spéciale

CONTRAINTES ÉCONOMIQUES :

- **GRUYÈRE** 40x1 + 20x2 + 50x3 < 3000 * 110
- **JAMBON** 50x1 + 0x2 + 60x3 < 3100 * 110
- **CHAMPIGNONS** 20x1 + 40x2 + 20x3 < 2000 * 80
- **LARDONS** 0x1 + 30x2 + 40x3 < 1000 * 70

CONTRAINTES DE SIGNE :

XI, X2,X3 > 0

FONCTION ÉCONOMIQUE :

Max z = XI +2X2 + 2X3

VARIABLES GARNITURES :

- y1 = grammage gruyère
- y2 = grammage jambon
- y3 = grammage champignons
- y4 = grammage lardons

CONTRAINTES BÉNÉFIQUES :

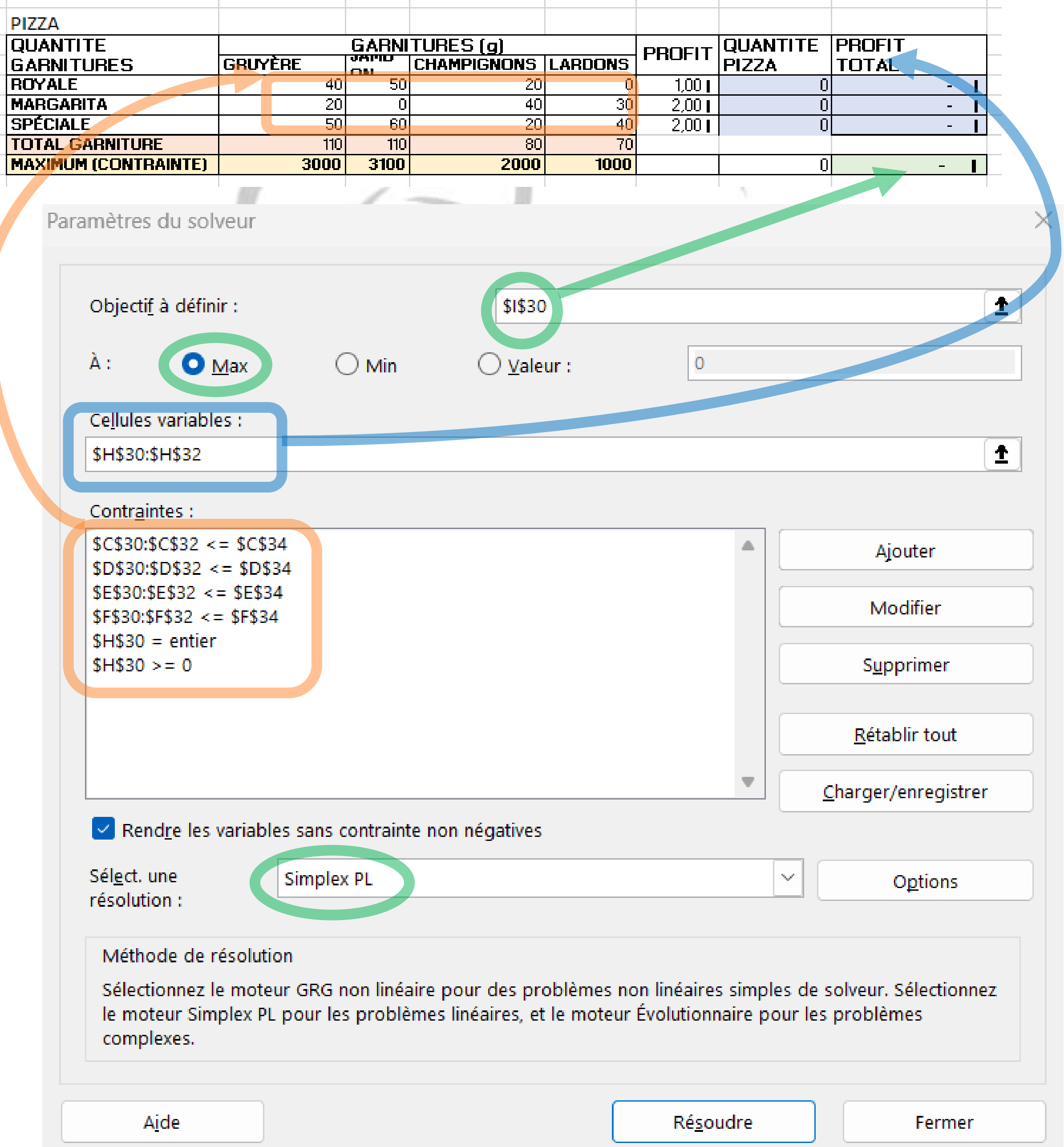
- 40y1 + 50y2+ 20y3 > 1
- 20y1 + 0y2 + 40y3 + 30y4 > 2
- 50y1 + 60y2 + 20y3 + 40y4 > 2

Min w = 3000y1 + 3100y2 + 2000y3 +1000 y4

y1, y2, y3, y4 > 0

Ce programme linéaire a trois variables d'optimisation, x1, x2 et x3, qui représentent respectivement le nombre de pizzas Royale, Margarita et Spéciale à fabriquer. Les contraintes du programme linéaire garantissent que la quantité totale de chaque ingrédient utilisé ne dépasse pas la quantité disponible.

Pour résoudre ce programme linéaire, on peut utiliser un solveur linéaire.



Le solveur linéaire donne la solution suivante :

- `x1 = 0`
- `x2 = 0`
- `x3 = 100`

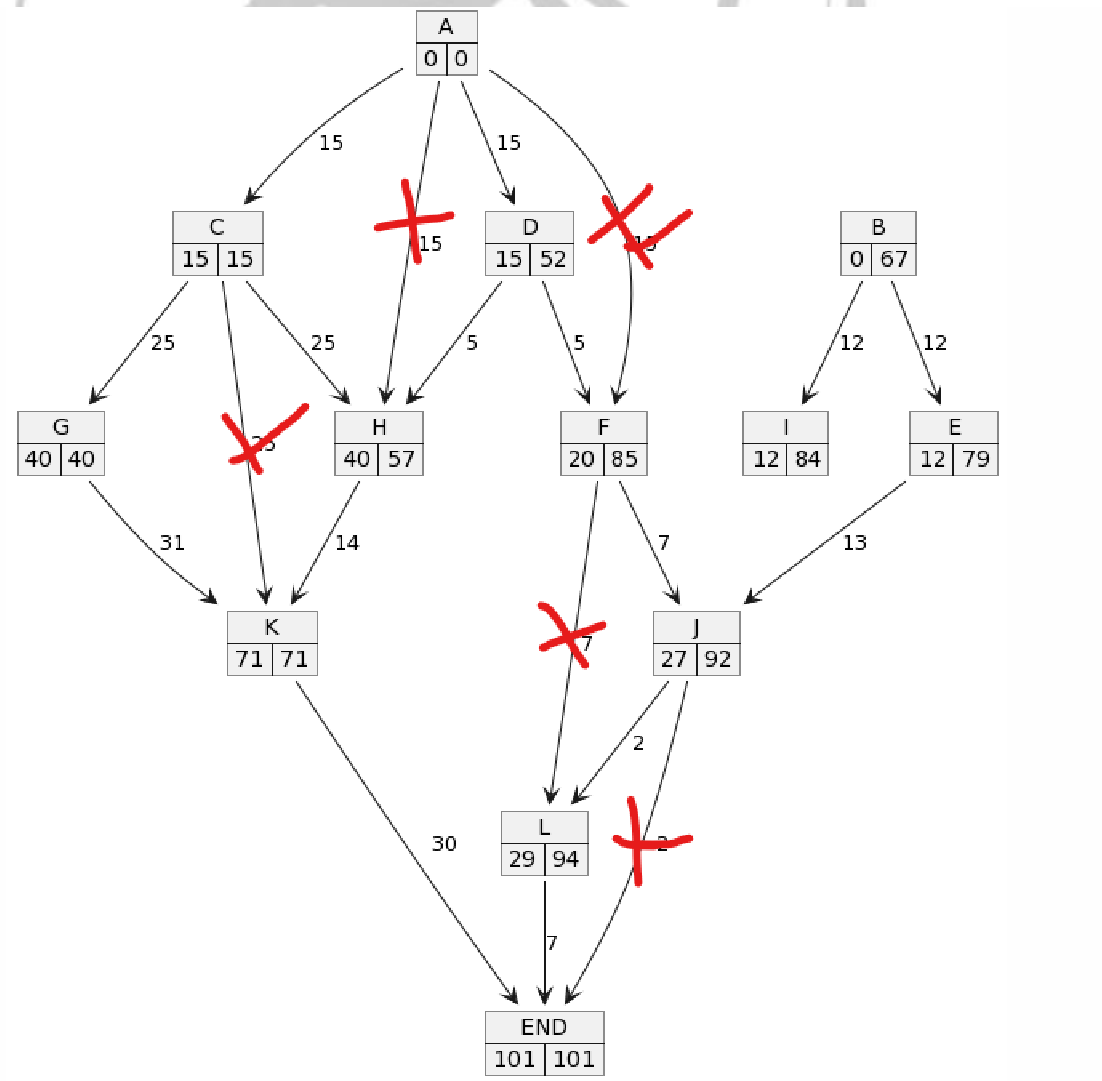
Cette solution signifie qu'il faut fabriquer 0 pizzas Royale, 0 pizzas Margarita et 100 pizzas Spéciale. Ce qui maximisera le chiffre d’affaire à 2.000 €.

PROBLÈME 3

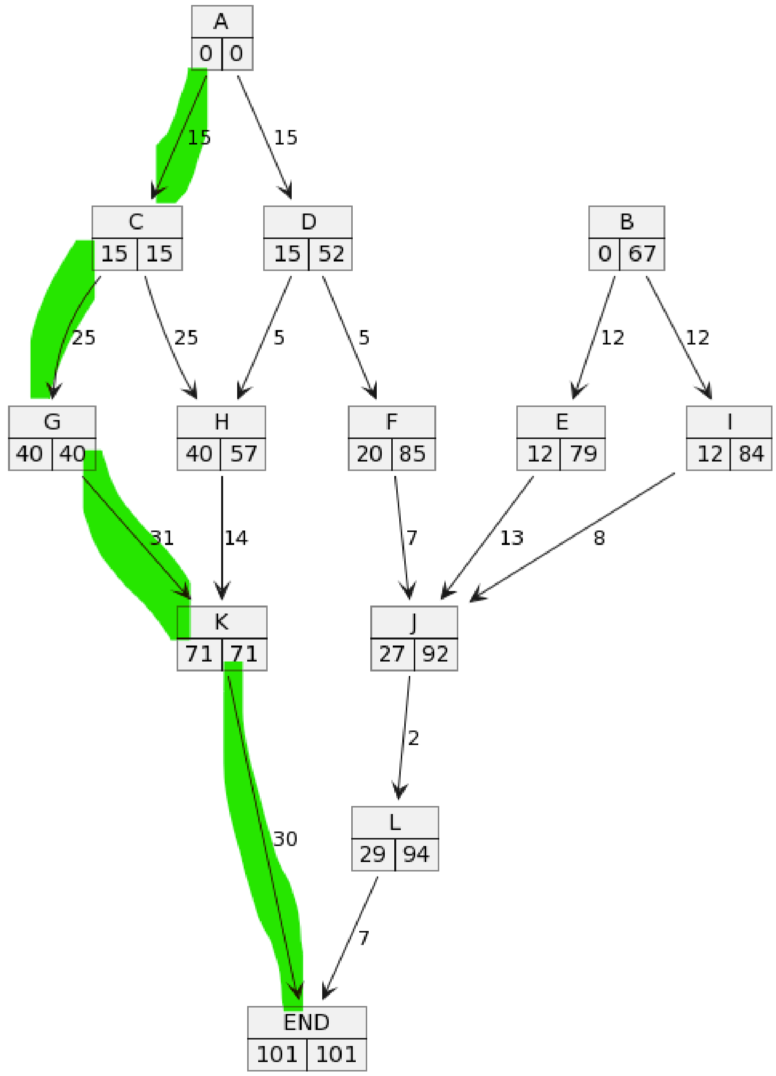
Le tableau ci-dessous précise les contraintes d'antériorité et la durée d'exécution en jours de chacune des tâches à effectuer avant l'ouverture de la succursale.

Tâches	Tâches antérieures	Durée
A	-	15
B	-	12
C	A	25
D	A	5
E	B	13
F	A, D	7
G	C	31
H	A, C, D	14
I	B	8
J	E, F, I	2
K	C, G, H	30
L	F, J	7

Utilisons les données fournies pour créer un diagramme réseau via PlantUML :



En supprimant les chemins redondants, nous obtenons ceci :



Dates au plus tôt et au plus tard ;

Tâches	Date au plus tôt	Date au plus tard
A	0	0
B	0	67
C	15	15
D	15	52
E	12	79
F	20	85
G	40	40
H	40	57
I	12	84
J	27	92
K	71	71
L	29	94

Résultats :

Durée totale du projet : 101 jours (calculée à partir de la date au plus tôt de la dernière tâche)

```
A -> C -> J -> K
```

Cela signifie que toutes les tâches critiques doivent être complétées selon le calendrier pour que le projet soit terminé en 101 jours.