Solution de la question 1:

In this problem, Benoît wants to maximize the number of points of his team, given some constraints. First, the decision variables are the different candidates :

$$\begin{aligned} x_1 &= Anakin & x_{11} &= Luke \\ x_2 &= Chewbacca & x_{12} &= Obi-Wan \\ x_3 &= Dooku & x_{13} &= Padme \\ x_4 &= Gial & x_{14} &= Qui-Gon \\ x_5 &= Han & x_{15} &= Sebulba \\ x_6 &= Jabba & x_{16} &= Sheev \\ x_7 &= Jubnuk & x_{17} &= Teebo \\ x_8 &= Lando & x_{18} &= Watto \\ x_9 &= Leia & x_{19} &= Yoda \\ x_{10} &= Lorth & x_{20} &= Zuckuss \end{aligned}$$

Let c_i denote the number of points of candidate i. Since Benoît wants to maximize the number of points in his team, we get the objective function

$$\max_{x\in\{0,1\}^{20}} \sum_{i=1}^{20} c_i x_i$$

Then, we have to model each constraint :

1. The team must contain at least 3 chefs who are skilled in preparing appetizers :
$$x_5 + x_9 + x_{11} + x_{13} + x_{19} \geq 3.$$
2. The team must contain at least 4 chefs who are skilled in preparing fish :
$$x_1 + x_2 + x_6 + x_7 + x_9 + x_{12} + x_{14} + x_{19} \geq 4.$$
3. The team must contain at least 4 chefs who are skilled in preparing meat :
$$x_1 + x_2 + x_3 + x_{12} + x_{15} + x_{16} + x_{17} + x_{18} \geq 4.$$

4. The team must contain at least 3 chefs who are skilled in preparing dessert.
$$x_3 + x_4 + x_8 + x_{10} + x_{20} \geq 3.$$
5. To promote young people, the team must contain at least 2 chefs who are 20 or less.
$$x_8 + x_9 + x_{14} + x_{18} \geq 2.$$
6. Yoda and Obi-Wan cannot stand each other, and it is not possible to have both of them in the team.
$$x_{12} + x_{19} \leq 1.$$
7. Han and Sheev have so much experience in working together that if one of the two is included in the team, the other must be too.
$$x_5 = x_{16}.$$
8. For the sake of fairness, not more than 3 chefs from the same restaurant should be hired in the same team. We observe that 4 candidates come from Crissier but this is the only restaurant with more than 3 chefs.
$$x_3 + x_5 + x_6 + x_{16} \leq 3.$$

Thus, the optimization problem is

$$\max_{x \in \{0,1\}^{20}} \sum_{i=1}^{20} c_i x_i$$

subject to

$$\begin{aligned}
x_5 + x_9 + x_{11} + x_{13} + x_{19} &\geq 3 \\
x_1 + x_2 + x_6 + x_7 + x_9 + x_{12} + x_{14} + x_{19} &\geq 4 \\
x_1 + x_2 + x_3 + x_{12} + x_{15} + x_{16} + x_{17} + x_{18} &\geq 4 \\
x_3 + x_4 + x_8 + x_{10} + x_{20} &\geq 3 \\
x_8 + x_9 + x_{14} + x_{18} &\geq 2 \\
x_{12} + x_{19} &\leq 1 \\
x_5 &= x_{16} \\
x_3 + x_5 + x_6 + x_{16} &\leq 3 \\
x_i &\in \{0,1\} \quad \forall i = 1, ..., 20.
\end{aligned}$$

The optimal solution of the problem is the one that includes all the candidates except Douku and Yoda.

Solution de la question 2:

We want to schedule the workshop employees in order to maximize their satisfaction according to the stated preferences. Thus, we define the :

— **Decision variables** : binary decision variables $x_{i,j,k}$ which are 1 if employee i works during the time slot j at day k and 0 otherwise, for $i = 1, 2, 3, 4,\ j = 1, ..., 8$ and $k = 1, ..., 5$.

— **Objective function** :
We denote by $a_{i,j,k}$ the reported preference of employee i for the time slot j at day k.
Since the goal is to maximize the satisfaction of the employees, we end up with the objective function

$$\max_x \sum_{i=1}^{4}\sum_{j=1}^{8}\sum_{k=1}^{5} a_{i,j,k}x_{i,j,k}$$

— **Constraints** :

— Each time slot must be covered by exactly one employee :

$$\sum_{i=1}^{4} x_{i,j,k} = 1, \quad \forall j \in \{1, \dots, 8\}\, \forall k \in \{1, \dots, 5\}\,.$$

— If an employee is not available for a time slot ⇒ she/he cannot be assigned to this time slot.

$$x_{i,j,k} \leq a_{i,j,k}, \quad \forall i \in \{1, \dots, 4\}\, \forall j \in \{1, \dots, 8\}\, \forall k \in \{1, \dots, 5\}\,.$$

— Every person must take a lunch break either between 12 :00 and 13 :00, or between 13 :00 and 14 :00 :

$$x_{i,4,k} + x_{i,5,k} \leq 1, \quad \forall i \in \{1, \dots, 4\}\, \forall k \in \{1, \dots, 5\}\,.$$

— Because of the noisy work environment, every person can work only two consecutive time slots. A break of at least one hour must be taken after that. That means if we consider for an employee i

at day k three consecutive time slots j, $j+1$ and $j+2$, we cannot assign the employee to at least one of these time slots :
$x_{i,j,k} = 0$ or $x_{i,j+1,k} = 0$ or $x_{i,j+2,k} = 0$, $\forall i \in \{1,\ldots,4\}\, \forall j \in \{1,...,6\}\, \forall k \in \{1,\ldots,5\}$
which leads to :

$$x_{i,j,k}+x_{i,j+1,k}+x_{i,j+2,k} \leq 2, \quad \forall i \in \{1,\ldots,4\}\, \forall j \in \{1,...,6\}\, \forall k \in \{1,\ldots,5\}\,.$$

— No one can work more than 20 hours per week :

$$\sum_{j=1}^{8}\sum_{k=1}^{5} x_{i,j,k} \leq 20, \quad \forall i \in \{1,\ldots,4\}\,.$$

Thus, we end up with the following binary linear optimization problem

$$\max_x \sum_{i=1}^{4}\sum_{j=1}^{8}\sum_{k=1}^{5} a_{i,j,k}x_{i,j,k}$$

subject to

$$\begin{aligned}
&\sum_{i=1}^{4} x_{i,j,k} = 1, && \forall j \in \{1,\ldots,8\}\, \forall k \in \{1,\ldots,5\}\\
&x_{i,j,k} \leq a_{i,j,k}, && \forall i \in \{1,\ldots,4\}\, \forall j \in \{1,\ldots,8\}\, \forall k \in \{1,\ldots,5\}\\
&x_{i,4,k} + x_{i,5,k} \leq 1, && \forall i \in \{1,\ldots,4\}\, \forall k \in \{1,\ldots,5\}\\
&x_{i,j,k} + x_{i,j+1,k} + x_{i,j+2,k} \leq 2, && \forall i \in \{1,\ldots,4\}\, \forall j \in \{1,...,6\}\, \forall k \in \{1,\ldots,5\}\\
&x_{i,j,k} \in \{0,1\}\,, && \forall i \in \{1,\ldots,4\}\, \forall j \in \{1,\ldots,8\}\, \forall k \in \{1,\ldots,5\}
\end{aligned}$$

There are several optimal solutions for this problem with the optimal value of the objective function $f^* = 211$. The decision variables $x_{i,j,k}$ of one optimal solution are given in the table below :

Solution de la question 3:

1. Un problème d'optimisation linéaire binaire est un problème dans lequel la fonction objectif ainsi que les contraintes sont linéaires et toutes les variables de décision $\in \{0,1\}$.

i	j	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
1	1	1	0	0	1	1
1	2	0	0	0	1	1
1	3	0	0	1	0	0
1	4	0	0	0	0	0
1	5	1	1	1	1	1
1	6	1	0	0	0	0
1	7	0	0	0	0	0
1	8	0	0	0	0	0
2	1	0	0	0	0	0
2	2	0	0	0	0	0
2	3	0	0	0	0	0
2	4	1	0	1	0	0
2	5	0	0	0	0	0
2	6	0	1	0	1	1
2	7	0	1	0	1	1
2	8	0	0	1	0	0
3	1	0	1	1	0	0
3	2	0	1	1	0	0
3	3	0	0	0	1	1
3	4	0	0	0	0	0
3	5	0	0	0	1	1
3	6	0	0	0	0	0
3	7	1	0	0	0	0
3	8	1	0	0	0	0
4	1	0	0	0	0	0
4	2	1	0	0	0	0
4	3	1	1	0	0	0
4	4	0	1	0	1	1
4	5	0	0	0	0	0
4	6	0	0	1	0	0
4	7	0	0	1	0	0
4	8	0	1	0	1	1

Dans ce problème, la fonction objectif et les contraintes sont linéaires. Il suffit de transformer les variable x_1 et x_2 en des variables booléennes. Ceci est possible car ces variables de décision prennent un nombre fini de valeurs. On définit :

> $x_1 = 1y_1 + 4y_2 + 6y_3$, avec $y_i \in \{0, 1\}$. Ici, au plus un y_i peut prendre la valeur 1 afin de pouvoir représenter toutes les valeurs que x_1 peut prendre. Si tout les y_i sont nuls alors la valeur de x_1

est 0.

$x_2 = -4y_4 - 2y_5 + 9y_6$, avec $y_i \in \{0, 1\}$. Ici, exactement un y_i peut prendre la valeur 1 afin de pouvoir représenter toutes les valeurs que x_2 peut prendre. x_2 ne prend pas la valeur 0.

et on remplace x_1 et x_2 par leur expression. On obtient donc :

$$
\begin{aligned}
& \max y_1 + 4y_2 + 6y_3 - 8y_4 - 4y_5 + 18y_6 \\
\text{s.c.} & \\
& y_1 + 4y_2 + 6y_3 - 4y_4 - 2y_5 + 9y_6 \leq 8 \\
& - y_1 - 4y_2 - 6y_3 - 4y_4 - 2y_5 + 9y_6 \leq 2 \\
& y_1 + 4y_2 + 6y_3 + 4y_4 + 2y_5 - 9y_6 \leq 4 \\
& y_1 + y_2 + y_3 \leq 1 \\
& y_4 + y_5 + y_6 = 1 \\
& y_i \in \{0, 1\}, \qquad \forall i = 1, ..., 6
\end{aligned}
$$

2. Afin de pouvoir représenter la nouvelle fonction objectif $\max x_1^2 + 2x_2$, on doit représenter toutes les valeurs que prend x_1^2. Puisque x_1 prend les valeurs 0, 1, 4 ou 6 alors x_1^2 prend les valeurs 0, 1, 16 ou 36. On définit donc :
   $x_1^2 = 1y_1 + 16y_2 + 36y_3$, avec $y_i \in \{0, 1\}$. De même que pour x_1, au plus un y_i peut prendre la valeur 1 afin de pouvoir représenter toutes les valeurs que x_1^2 peut prendre. Si tout les y_i sont nuls alors la valeur de x_1^2 est 0.
   Ainsi, la fonction objective est modifiée de la manière suivante :

$$\max y_1 + 16y_2 + 36y_3 - 8y_4 - 4y_5 + 18y_6$$

Solution de la question 4:

1. Afin de modéliser ce problème comme un problème d'optimisation, nous définissons :
   — **Variables de décision :**
      — x_1 nombre d'armoires produites par l'ébéniste.

  — x_2 nombre de tables produites par l'ébéniste.

— **Fonction objectif :** l'ébéniste cherche à maximiser son profit. On a donc :

$$\max 8x_1 + 5x_2$$

— **Les contraintes :**
Le nombre total d'heures de travail ne doit pas dépasser $6h$.

$$x_1 + x_2 \leq 6$$

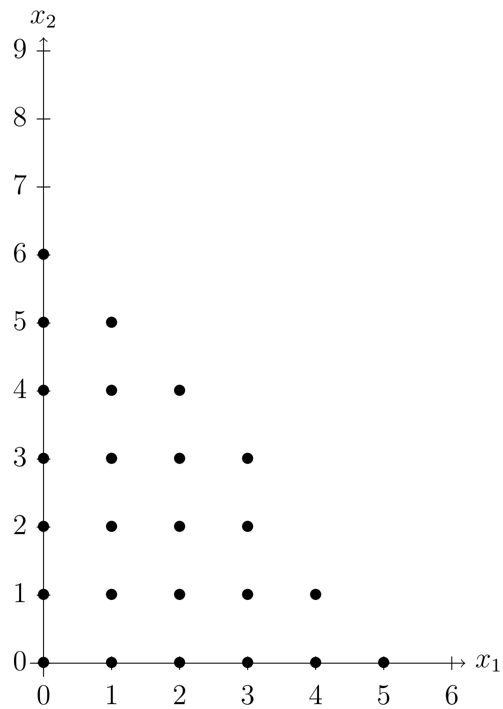
Le nombre total de m^2 de bois utilisé ne doit pas dépasser $45h$.

$$9x_1 + 5x_2 \leq 45$$

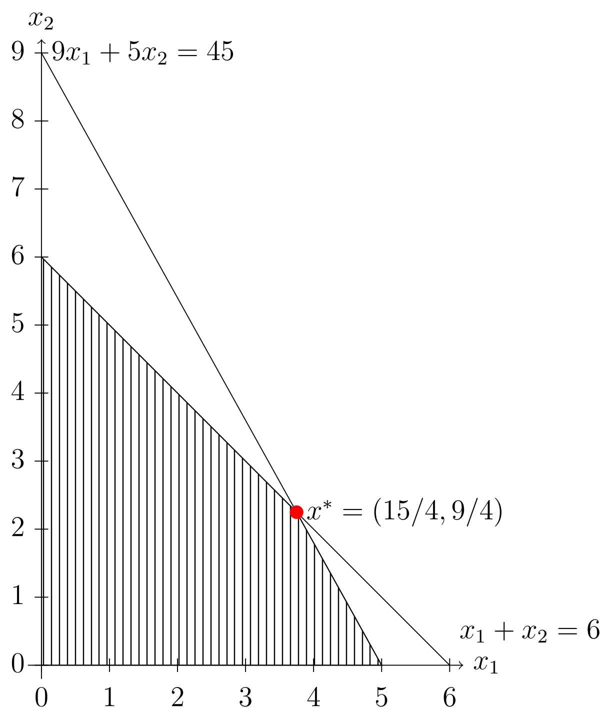
Nous obtenons donc le problème d'optimisation linéaire en nombre entier suivant :

$$\begin{aligned} & \max 8x_1 + 5x_2 \\ \text{s.c :} & \\ & x_1 + x_2 \leq 6 \\ & 9x_1 + 5x_2 \leq 45 \\ & x_i \in \mathbb{N} \end{aligned}$$

2. Le domaine admissible du problème P est représenté par la figure suivante :



3. On résout graphiquement le problème $R(P)$. On obtient le domaine admissible pour $R(P)$ sur la figure suivante :



La solution optimale est donnée par : $x^* = (15/4, 9/4)$ avec la valeur optimale de la fonction objectif $f^* = 41.25$.

Si cette solution était entière, elle serait aussi la solution optimale du problème original P. Ici ce n'est pas le cas, mais la valeur f^* est une borne supérieure pour la valeur optimale de la fonction objectif du problème P.

4. L'objectif est de résoudre le problème P par la méthode de séparation/évaluation. Pour cela, on définit :
   — **La règle de séparation :** Si la solution optimale x_R^* de la relaxation linéaire du problème $R(P_k)$ n'est pas entière alors on choisit une composante non entière $(x_R^*)_i$ et on crée les 2 sous-problèmes suivants :
      — Le sous-problème P_k^l en rajoutant la contrainte suivante

$$x_i \leq \lfloor (x_R^*)_i \rfloor$$

.

— Le sous-problème P_k^r en rajoutant la contrainte suivante

$$x_i \geq \lceil (x_R^*)_i \rceil$$

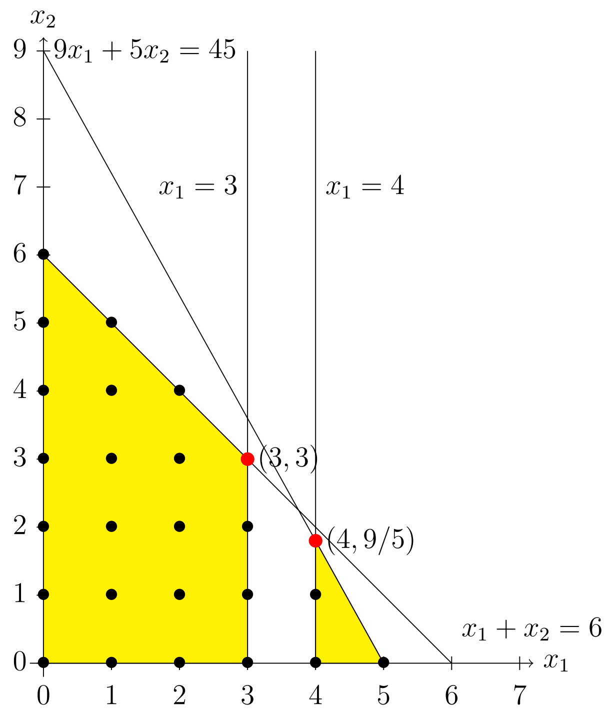
.

- **Une borne inférieure :** On initialise la borne inférieure à $-\infty$ puisqu'on ne dispose pas d'une solution admissible du problème.
- **Une borne supérieure :** Une borne supérieure du problème P_k est la valeur optimale de la fonction objectif du problème $R(P_k)$.
- **La stratégie de parcours de l'arbre :** On parcourt l'arbre en profondeur d'abord. Si plusieurs noeuds se trouve à la même profondeur dans l'arbre. On choisit celui qui a la meilleur borne supérieure.

Appliquons cette méthode pour notre problème. La solution optimale du problème $R(P)$ est $x^* = (15/4, 9/4)$. Les deux composantes de la solution étant non entière, nous choisissons arbitrairement de faire la séparation sur la variable x_1. Nous obtenons les 2 sous-problèmes suivants :

- P_1 en rajoutant la contrainte suivante $x_1 \leq 3$ au problème P.
- P_2 en rajoutant la contrainte suivante $x_1 \geq 4$ au problème P.

Ces 2 contraintes sont représentées sur la figure suivante :



On résout graphiquement la relaxation linéaire de P_1 et de P_2 :

— La solution optimale de $R(P_1)$ est $x^* = (3, 3)$ et la valeur optimale de la fonction objectif est $f^* = 39$. La solution étant entière, il s'agit d'une solution admissible du problème originale P. On peut donc mettre à jour la valeur de la borne inférieure qui passe de $-\infty$ à 39. La borne supérieure de P_1 est 39. De plus, on n'a pas besoin d'explorer ce noeud (on a pas besoin de créer des noeuds fils de ce noeud).

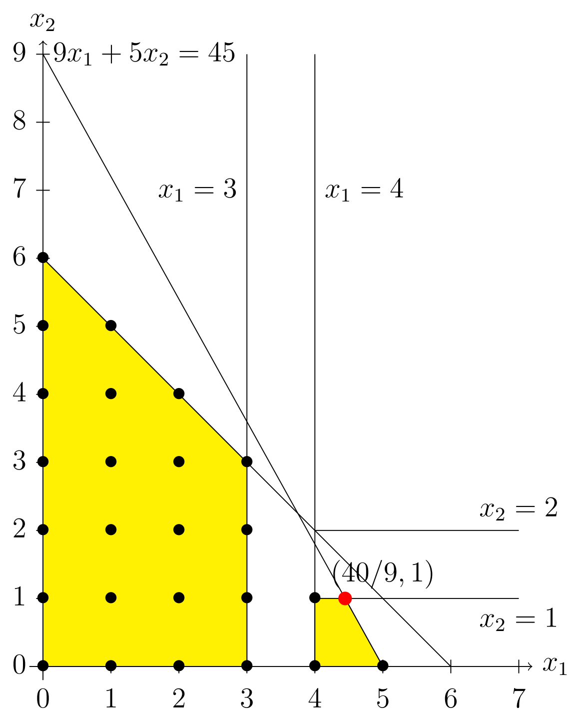
— La solution optimale de $R(P_2)$ est $x^* = (4, 9/5)$ et la valeur optimale de la fonction objectif est $f^* = 41$. La borne supérieure de P_2 est donc 41.

A présent, on crée les sous problèmes de P_2 en faisant la séparation sur la variable non entière $x_2 = 9/5$. On obtient les deux sous problèmes suivants :

— P_{21} en rajoutant la contrainte suivante $x_2 \leq 1$ au problème P_2.

— P_{22} en rajoutant la contrainte suivante $x_2 \geq 2$ au problème P_2.

Ces 2 contraintes sont représentées sur la figure suivante :



— La solution optimale de $R(P_{21})$ est $x^* = (40/9, 1)$ et la valeur optimale de la fonction objectif est $f^* = 365/9$. La borne supérieure de P_{21} est donc $365/9$.

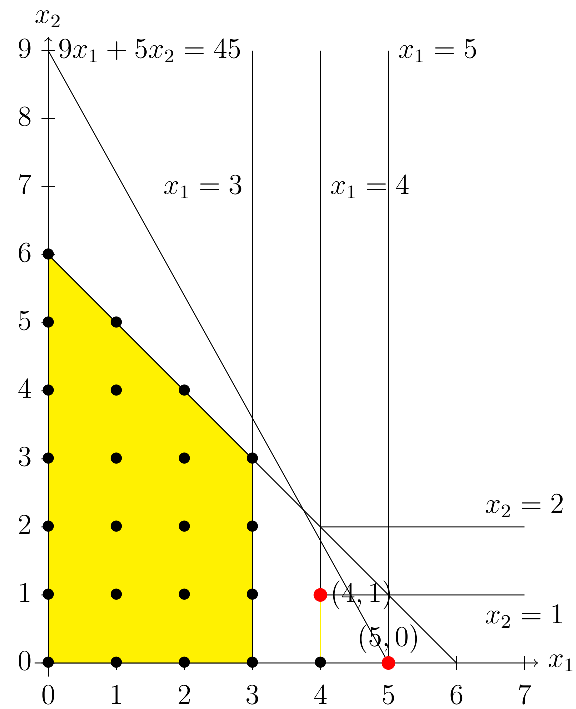
— Le problème P_{22} est non réalisable. Ce noeud est donc éliminé.

A présent, on crée les sous problèmes de P_{21} en faisant la séparation sur la variable non entière $x_1 = 40/9$. On obtient les deux sous problème suivants :

— P_{211} en rajoutant la contrainte suivante $x_1 \leq 4$ au problème P_{21}.

— P_{212} en rajoutant la contrainte suivante $x_1 \geq 5$ au problème P_{21}.

On rajoute la dernière contrainte $x_1 \geq 5$ sur la figure suivante :



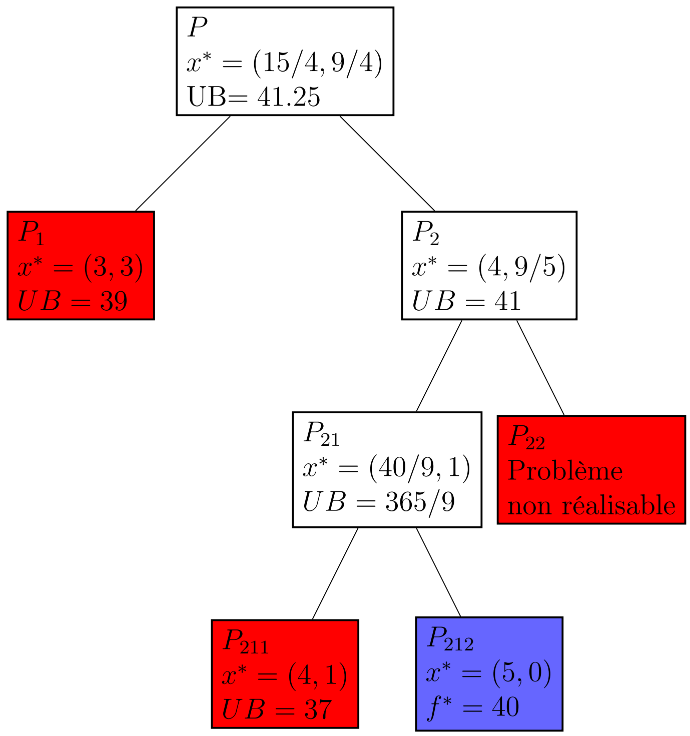
On résout graphiquement la relaxation linéaire de P_{211} et P_{212} :

— La solution optimale de $R(P_{211})$ est $x^* = (4, 1)$ et la valeur optimale de la fonction objectif est $f^* = 37$. La borne supérieure de P_{211} est 37. Cette borne étant inférieure à la borne inférieure actuelle. Ce noeud est donc éliminé de l'arbre.

— La solution optimale de $R(P_{212})$ est $x^* = (5, 0)$ et la valeur optimale de la fonction objectif est $f^* = 40$. La solution étant entière, il s'agit d'une solution optimale du problème originale P. On peut donc mettre à jour la valeur de la borne inférieure qui passe de 39 à 40. Le noeud P_1 est donc éliminé car sa borne supérieure est inférieur à 40.

On a exploré tout l'arbre de séparation/évaluation. Ainsi, la solution optimale est donnée par le noeud P_{212} qui correspond à $x^* = (5, 0)$ et $f^* = 40$.

L'exécution de l'algorithme est donné par l'arbre suivant, où les noeuds éliminés sont représentés en rouge et la solution optimale est représenté en bleu :



Solution de la question 5:

1. It is a maximization problem, since the upper bound (the solution of the relaxed problem) decreases as we go down in the tree. In the beginning (the root) we are too optimistic and the upper bound is high. As we add constraints we get closer to the optimal solution and the objective function value decreases.
2. The current optimal solution must be integer with the largest objective value and thus is $x_4^\star$ with optimal value 7.
3. The nodes which can be discarded are the ones corresponding to infeasible solutions, and all nodes for which the upper bound is lower than 7 (since it is the optimal value at this step of the algorithm). Here, P_1, P_3 and P_6 can be discarded.
4. The algorithm is not terminated since $x_5^\star$ is not integer and the upper bound of P_5 is greater than $f(x_4^\star)$. We must create subproblems of problem P_5 by adding the suitable constraints.