



## Lycée Français Vincent van Gogh, Pays Bas

Atelier 2023

Elèves : Justine Seguy, Solène Debroux, Joséphine Vermelle, Anastasiia Berezovska, Golda Majerowicz

Niveau: 1ere

Professeurs animateurs: M. Beringue, Mme. Boissonnet, M. Buchwald et Mme. Decool

Chercheuse : Marie Anastacio, *RWTH university*

# SALLE D'EXAMEN





## **Sommaire:**

1. Introduction
2. Une première approche: le traitement de donnée
3. Le développement d'un algorithme
4. Programmer l'algorithme en python
5. Programmer l'algorithme en Minizinc
6. Conclusion

### **I. Introduction:**

Au Lycée Français Vincent van Gogh, nous sommes confrontées à un problème mathématique captivant qui nous a été soumis par la chercheuse Marie Anastacio. Notre défi réside dans l'optimisation du temps et de l'espace pour organiser la passation de plusieurs examens, en prenant en compte à la fois la disponibilité des salles et le nombre d'élèves concernés.

Pour commencer, il y a plusieurs contraintes à prendre en compte pour la résolution de ce problème. La première étant le fait qu'il est interdit de faire passer deux examens différents dans la même salle. De plus, certains élèves doivent passer plusieurs examens, il est donc crucial que chaque élève puisse passer tous ses examens. Il faut ainsi s'assurer qu'aucun élève ne participe à deux examens en même temps. Enfin, il faut parvenir à optimiser au maximum l'emploi du temps, de sorte à condenser la période des examens sur la plus petite durée.

### **II. Traitement de données:**

La chercheuse qui nous accompagnait nous a fourni une liste écrasante de données. Nous étions confrontés à une liste de 800 disciplines et 7689 élèves, chaque élève passait en moyenne 4 examens, pour un total de 36000 déroulements. La chercheuse nous a également transmis une liste de 16 salles avec des capacités variant de 35 à 200 places.

Par conséquent, la première étape fut le traitement de données. Nous avons utilisé Google Sheets afin de les trier de différentes façons.

La première méthode à été d'organiser les examens en fonction du nombre d'élèves inscrits à chaque discipline. Comme il est présenté ci-dessous:



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	D940031575	B11102E1							HGAEM2E1	542	HGAEM2E1	542	
2	D940031645	B11102E1							HGBEM3E1	340	HGBEM3E1	340	
3	A945757865	AAA033E1							G111GBE1	222	G111GBE1	222	
4	B940035469	AAA011E1							C71101E1	205	C71101E1	205	
5	D940029697	B31103E1							R11001E1	191	R11001E1	191	
6	D940036112	B31101E1							M3A010E1	187	M3A010E1	187	
7	D940037094	B31101E1							M3B030E1	184	M3B030E1	184	
8	D940036400	B31101E1							M3A015E1	176	M3A015E1	176	
9	A936219093	AAA024E1							F311T1E1	175	F311T1E1	175	
10	A935096656	AAA034E1							G11MA1E1	172	G11MA1E1	172	
11	D940033030	B31101E1							M31001E1	171	M31001E1	171	
12	D940033591	B31101E1							M31012E1	168	M31012E1	168	
13	A920200351	AAA034E1							G111VAE1	165	G111VAE1	165	
14	A935233982	AAA034E1							G11PR1E1	165	G11PR1E1	165	
15	A922534920	AA3008E1							C11105E1	163	C11105E1	163	
16	A935215885	AAA024E1							F311M1E1	161	F311M1E1	161	
17	D937220070	B12302E1							L11100E1	160	L11100E1	160	
18	A936041416	AAA024E1							F311Q1E1	157	F311Q1E1	157	
19	D940036020	B31101E1							M3A002E1	155	M3A002E1	155	

Les examens sont triés en ordre décroissant selon le nombre d'élèves qui y participent.

La deuxième méthode a été d'organiser les examens en fonction des disciplines choisies par l'élève. Comme vous pouvez le voir ci-dessous:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	AA3008E1	HOLLYWOOD &	03.00		AAB025E1	R13001E1		A920320659	AA2016E1				
2	AAB025E1	INTRODUCTION	03.00		A890186790	R13006E1		A920359763	AA2016E1				
3	B11101E1	FUNDAMENTAL	03.00		A890186790	R13016E1		A921678313	AA2016E1				
4	C12332E1	ENVIRONMENTAL	03.00		A890186790	R13021E1		A927642828	AA2016E1				
5	C13562E1	PARASITE IMMI	03.00		A890186790	R13022E1		A928203460	AA2016E1				
6	C13563E1	MOLECULAR &	03.00		A891097581	R23102E1		A930504647	AA2016E1				
7	C13564E1	ECOTOXICOLO	03.00		A891097581	R23107E1		A930549305	AA2016E1				
8	C13565E1	EVOLUTION & É	03.00		A891097581	R23111E1		A930729996	AA2016E1				
9	C13567E1	APPLIED MICR	03.00		A891097581	R23117E1		A931502705	AA2016E1				
10	C13569E1	FUNDAMENTAL	03.00		A891097581	R23209E1		A935254493	AA2016E1				
11	C13573E1	EVOLUTIONAR	03.00		A892884191	Q4B102E1		A935319035	AA2016E1				
12	C43511E1	GENETICS AND	03.00		A892884227	AA3008E1		A935508961	AA2016E1				
13	C43512E1	GENE CONTRC	03.00		A892884294	V73232E1		A935520158	AA2016E1				
14	C72303E1	PROTEIN & GEI	03.00		A892884294	V73MD1E1		A935611009	AA2016E1				
15	C73507E1	ADVANCED BIC	03.00		A892884308	Q3A308E1		A936074182	AA2016E1				
16	C8P001E1	MSC OCC PSY	03.00		A892884308	V62200E1		A936910710	AA2016E1				
17	C8P002E1	MSC OCC PSY	03.00		A892884319	V73128E1		A936920904	AA2016E1				
18	G11MA1E1	INTRODUCTOR	03.00		A892884331	V73232E1		A939969739	AA2016E1				
19	H3CDMSE1	DYNAMICS OF	03.00		A892884331	V73MD1E1		A940051746	AA2016E1				

En cherchant l'identifiant d'un élève en particulier, tous les examens auxquels il participe apparaissent.

Cependant, le traitement de données est loin d'être suffisant pour un problème de cette taille. C'est pourquoi nous nous sommes penchés vers l'utilisation d'un algorithme.

### III. Algorithme:

#### A. Insérer les données:

La première étape est d'insérer les données telles que le nombre de chaises par salle noté  $C_i$  (avec  $i$  le numéro de salle), le groupe d'élèves par matière noté  $G_n$  (avec  $n$  le numéro du groupe), et le nombre d'heures que va durer chaque examen par matière noté  $N_h$ .



## **B. Regrouper et classer:**

Ensuite, on regroupe et classe les  $G_n$  de manière décroissante, attribue à chaque  $G_n$  ses élèves et vérifie qu'il y a assez de place dans une salle; sinon on découpe le groupe en plusieurs sous-groupes différents dans des salles différentes ou/et dans un créneau horaire différent.

## **C. Vérification:**

Suite aux premières étapes, on vérifie les deux paramètres suivants:

Si un élève appartient à deux ou plus de  $G_n$  associés dans un même horaire, cela signifierait que l'élève doit être dans deux lieux distincts en même temps, ce qui n'est physiquement pas possible (*et, malheureusement, on n'a pas le droit de couper les élèves en deux*).

De plus, l'algorithme vérifie si deux examens différents ne se déroulent pas dans une même salle parce que cela est une de nos contraintes.

Si l'algorithme s'aperçoit qu'un élève est censé passer deux examens différents qui ont lieu au même moment, alors il va remplacer un de ces examens par un autre pour que tous les élèves puissent assister à leurs examens. Si cela n'est cependant pas possible, alors l'algorithme admet un créneau entier pour un de ces deux examens.

Si l'algorithme s'aperçoit qu'effectivement il y a deux examens distincts dans une même salle, il déplace l'un de ces examens dans une autre salle, sinon, dans un autre créneau horaire ou un autre jour.

Lorsque toutes les conditions citées précédemment seront vérifiées, l'algorithme vérifie s'il y a sept heures d'examen par jour et par salle. Si cela est le cas, alors il gardera les emplacements. Dans le cas contraire, s'il y a plus de sept heures d'examen par jour et par salle, il déplace les examens en trop le jour suivant.

## **D. Optimisation:**

Ainsi, on arrive à l'étape qui nous a posé le plus de problèmes : l'optimisation.

Pour commencer nous allons multiplier le nombre de  $H_{max}$  et  $E_{max}$  (Horaire Maximum par jour et le maximum d'Examens associés) qui nous donne le résultat nommé  $Th$ , qui correspond au total maximum des heures d'examens.

Ensuite nous allons faire la somme des  $N_h$ , noté  $Sh$ , de tous les  $G_n$  présents dans une coupe de 7h (= 1 jour). Nous allons à présent étudier la différence entre  $Th$  et  $Sh$ . Si  $Th$  équivaut à  $Sh$  alors on garde l'emploi du temps tel quel. En effet, si cette



égalité apparaît, alors cela signifie, qu'il y a autant d'heures d'examens prévues que d'heures possibles dans une journée ( $T_h$ ).

Ensuite une autre possibilité, est que  $T_h$  soit supérieur à  $S_h$ . Alors la soustraction  $T_h - S_h$  représente le nombre d'heures encore inoccupé. Ensuite on va essayer de trouver et d'associer un  $G_n$  qui correspond aux différentes contraintes (chaises, heures, élèves).

Représentation d' un jour ou l'optimisation est utile:

	$E_n$	$E_{n+1}$	...	$E_{max}$
$H_n$	G1	G2	G3	
$H_{n+1}$	G1		G3	
...			G3	
$H_{max}$				

Vertical dimension labels: 1h (for the first row), 7h (for the entire table height).

Avec:

$H$  : 1h

$E_n$  : un Examen

$S_h$  : la somme de tous les  $N_h$  d'une journée

$J$  : le nombre d'heures à combler

### **E. Organisation:**

Après l'optimisation, l'algorithme doit vérifier s'il y a assez de place pour tous les élèves. Pour cela il calcule le nombre total de chaises par salle disponibles dans tout un créneau horaire auquel il soustrait le nombre total de chaises occupées en ce même créneau horaire.

Si le résultat obtenu est supérieur ou égal à 0, alors l'algorithme garde l'emplacement.

Si le résultat obtenu est inférieur à 0, alors l'algorithme fait en sorte qu'il y a assez de place pour les élèves en déplaçant un des examens associés dans un autre créneau horaire.

### **F. Surveillants:**

Nous allons ensuite attribuer un surveillant par groupe de 25 élèves.

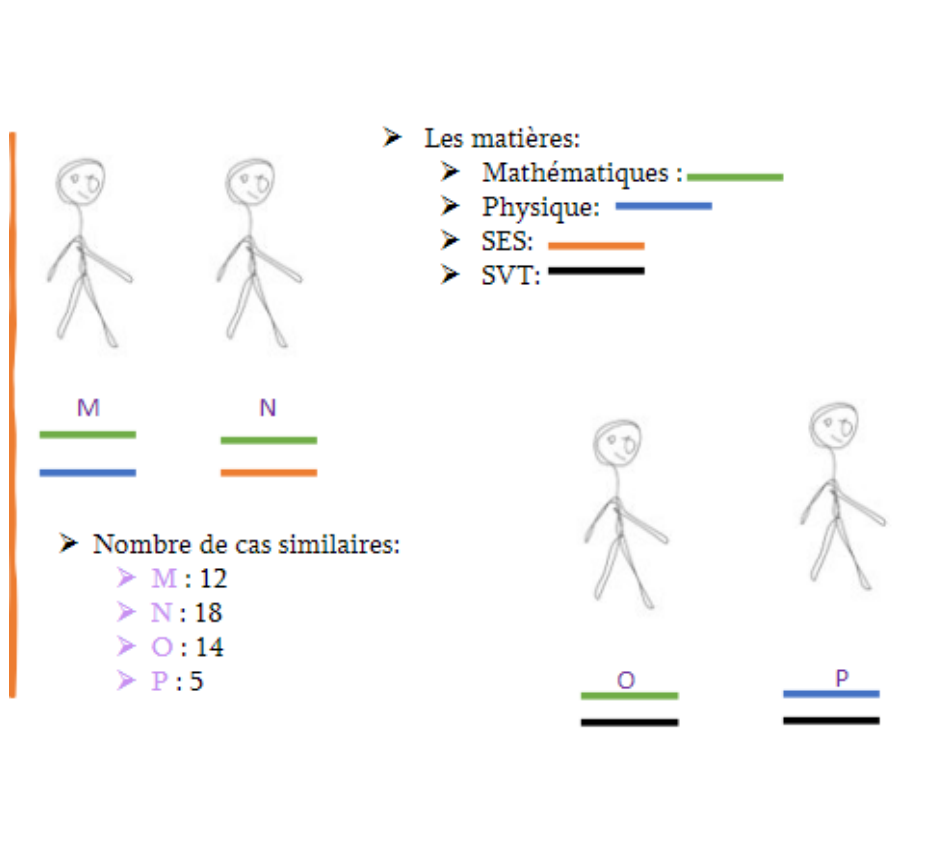


### G. Afficher les données:

Finalement, l'algorithme va afficher l'emploi du temps final avec la répartition des examens en fonction des salles, le nombre de jours nécessaires pour passer tous les examens, et le nombre de surveillants pour chaque examen.

### H. Exemple afin d'illustrer l'algorithme littéraire:

Pour illustrer notre algorithme, voici un exemple précis. Pour cela, prenons 4 spécialités: les mathématiques, les S.E.S, la S.V.T et la physique. Nous avons alors pris 49 élèves et 2 salles: 1 salle de 20 places et une de 15 places.



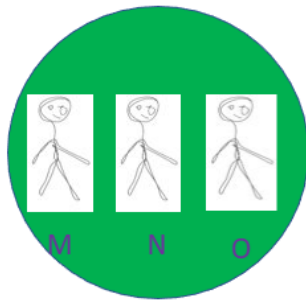
Nous allons ensuite créer des groupes d'élèves: M ; N ; O et P qui font les mêmes choix de spécialités. Ces groupes ont une taille aléatoire. Il y donc:

- Le groupe M, de 12 élèves qui font mathématiques et physiques,
- Le groupe N, de 18 élèves qui font mathématiques et S.E.S.
- Le groupe O, de 14 élèves qui font mathématiques et S.V.T.
- Le groupe P, de 5 élèves qui font physiques et S.V.T.

Premièrement, l'algorithme va former des groupes d'élèves selon les examens qu'ils doivent passer en additionnant les effectifs de chaque discipline.

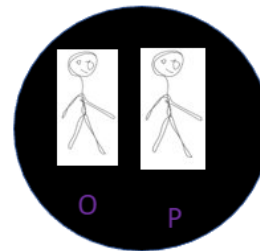


Dans le groupe mathématique il y aura donc 12+18+14 élèves ce qui donne un total de 44 élèves dans ce groupe. En poursuivant ce système, l'algorithme trouve alors que 19 élèves passent l'épreuve de S.V.T., 18 pour la S.E.S. et 17 pour la physique. L'algorithme va classer ces groupes dans l'ordre décroissant par la suite.



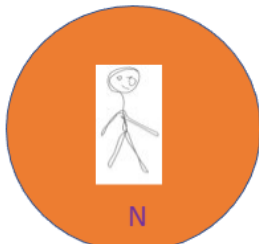
$$\begin{aligned} G_1 &= M+N+O \\ &= 12+18+14 \\ &= 44 \end{aligned}$$

G<sub>1</sub>: Maths



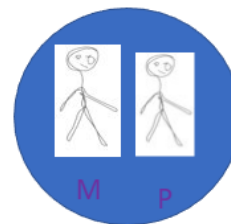
$$\begin{aligned} G_2 &= 14+5 \\ &= 19 \end{aligned}$$

G<sub>2</sub>: SVT



$$\begin{aligned} G_3 &= N \\ &= 18 \end{aligned}$$

G<sub>3</sub>: SES



$$\begin{aligned} G_4 &= M+P \\ &= 12+5 \\ &= 17 \end{aligned}$$

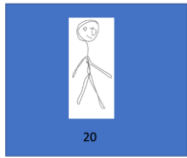
G<sub>4</sub>: Physique

La répartition dans les salles peut alors commencer. Pour cela, l'algorithme va placer les élèves dans les salles jour par jour. Pour y arriver, il va placer le plus gros groupe dans la plus grande salle, puis il va soustraire le nombre d'élèves du groupe au nombre de chaises qu'il y a dans la salle. Si le nombre est positif ou égale à 0 il va alors prendre le prochain groupe le plus grand et le mettre dans la deuxième salle ou dans sa plus grande salle le lendemain. En revanche, si la différence est négative, il va déplacer le reste des élèves dans la deuxième salle ou le lendemain. Et ainsi de suite jusqu'à ce qu'il n'y aie plus d'élèves à placer.

Voici le dérouler:

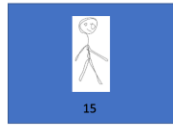


### Répartition des salles au jour 1



S<sub>1</sub>: 20 C<sub>1</sub>

$$C_1 - G_1 = X_1 \\ 20 - 44 = -24 < 0$$

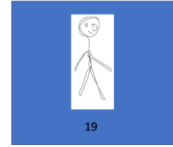


S<sub>2</sub>: 15 C<sub>2</sub>

$$C_2 - |X_1| = X_2 \\ 15 - 24 = -9 < 0 \text{ donc on continue.}$$

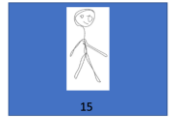
Office Online Frame

### Répartition des salles au jour 2



S<sub>1</sub>: 20 C<sub>1</sub>

$$C_1 - G_2 = X_4 \\ 20 - 19 = 1 > 0 \text{ donc finit}$$



S<sub>2</sub>: 15 C<sub>2</sub>

$$C_2 - G_3 = X_5 \\ 15 - 18 = -3 < 0$$

Jour 3

### Répartition des salles au jour 3



S<sub>1</sub>: 20 C<sub>1</sub>

$$C_1 - |X_1| = X_6 \\ 15 - 14 = 1 > 0 \text{ donc finit}$$

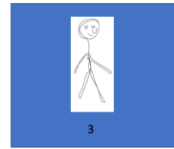


S<sub>2</sub>: 15 C<sub>2</sub>

$$C_2 - G_4 = X_7 \\ 15 - 18 = -3 < 0$$

Jour 4

### Répartition des salles au jour 4



S<sub>1</sub>: 20 C<sub>1</sub>

$$C_1 - |X_6| = X_7 \\ 20 - 3 = 17 > 0 \text{ donc finit}$$



S<sub>2</sub>: 15 C<sub>2</sub>

$$C_2 - G_8 = X_8 \\ 15 - 17 = -2 < 0$$

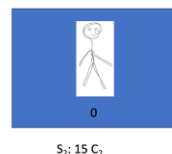
Jour 5

### Répartition des salles au jour 5



S<sub>1</sub>: 20 C<sub>1</sub>

$$C_1 - |X_8| = X_9 \\ 20 - 2 = 18 > 0 \text{ donc finit}$$



S<sub>2</sub>: 15 C<sub>2</sub>

Finalement nous avons mis les résultats sous forme de tableau comme pour un emploi du temps:





## Emploi du temps obtenus

Jours Numéro de salle	Jour 1	Jour 2	Jour 3	Jour 4	Jour 5
S1	G1: Mathématiques	G1: Mathématiques	G2: SVT	G3: SES	G4: Physique
	20 élèves dans la salle	1 élève dans la salle	4 élèves dans la salle	3 élèves dans la salle	2 élèves dans la salle
S2	G1: Mathématiques	G2: SVT	G3: SES	G4: Physique	
	15 élèves dans la salle	15 élèves dans la salle	15 élèves dans la salle	15 élèves dans la salle	

### IV. L'algorithme Python et Minizinc:

La prochaine étape était de transcrire cet algorithme en programme informatique.

La particularité du problème qui nous a été proposé est un problème de contraintes. Les notions qui sont abordées ne se traduisent pas directement sous forme de nombres. A partir du moment où l'on est humain, ces contraintes et la façon dont elles opèrent sont limpides et simples à comprendre, cependant ce n'est pas le cas pour un ordinateur.

Le premier langage essayé était Python car c'est le langage de programmation des lycéens.

- La première étape était de récolter les informations et de faire en sorte que l'algorithme classe les informations de telle sorte à pouvoir être traitées correctement.
- Puis il s'agit de définir un emploi du temps et comment fonctionnait ce dernier, c'est-à-dire définir un créneau horaire, un jour, et comment remplir chaque créneau horaire.

Le programme se déroule en deux parties:

- Dans un premier temps:
  - Classer les examens dans l'ordre décroissant du nombre d'élèves.
  - Placer dans la plus grande salle le maximum d'élèves du premier examen. Pour associer à chaque élève sa salle, prendre cette liste d'élèves dans l'ordre donné de cet examen et ajouter les élèves.



- Dans un second temps:
  - Le programme s'assure qu'aucun élève ne se retrouve deux fois dans le même créneau horaire:
  - Si le programme trouve qu'un élève se retrouve plusieurs fois dans le créneau horaire:
    - Déplacer l'ensemble salle-élèves à un créneau horaire où la salle est disponible, en favorisant les créneaux horaires se trouvant plus tôt dans la journée.

Lorsque l'on tourne le programme, il fournit une réponse satisfaisante, cependant est-ce la plus optimale?

Résultat: (prendre en compte que 1 examen = 1 créneau horaire)

```

Jour 1:
Créneau horaire 1: Maths - Salle: Salle A - Élèves: Valerie, Xavier, Yvonne,
Gerry, Clotho, Yorick, Gunther, Horadrick, Shepherd, Vagner, Cory, Beatrice,
Rosa, Omori, Marie, Queen, Ralf, Sebastien, Titouan, Ursula.
SES - Salle: Salle B - Élèves: Fred, Emily, Bobette, Celine, Gerard, Sallyface,
Hector, Ines, Josette, Kevin, Louise, Marnie, Nicole, Oriane, Patrick
Créneau horaire 2: Maths - Salle: Salle A - Élèves: Kel, Sandy, Leah, Abigail,
Sam, Eliot, Aubrey, Sunny, Hoagie, Laverne, Fred, Emily, Bobette, Celine, Gerard,
Sallyface, Hector, Ines, Josette Kevin
SES - Salle: Salle B - Élèves: Queen, Ralf, Sebastien, Titouan, Ursula, Valerie,
Xavier, Yvonne, Gerry, Clotho, Yorick, Gunther, Horadrick
Créneau horaire 3: Physiques - Salle: Salle A - Élèves: Alex, Blob, Charline,
Dave, Eve, Omori, Marie, Kel, Sandy, Leah, Abigail, Sam, Eliot, Aubrey, Sunny,
Hoagie, Laverne
SES - Salle: Salle B - Élèves: Shepherd, Vagner, Cory, Beatrice, Rosa
Créneau horaire 4: Maths - Salle: Salle A - Élèves: Louise, Marnie, Nicole, Oriane,
Patrick
SVT - Salle: Salle B - Élèves: Alex, Blob, Charline, Dave, Eve
  
```

Mis sous forme de tableau:

	Salle A (C = 20)	Salle B (C = 15)
H1	<b>Maths</b> (Valérie, Xavier, Yvonne, Gerry, Clotho, Yorick, Gunther, Horadrick, Shepherd, Vagner, Cory, Beatrice, Rosa, Omori, Marie, Queen, Ralf, Sebastien, Titouan, Ursula)	<b>SES</b> (Fred, Emily, Bobette, Céline, Gerard, Sallyface, Hector, Ines, Josette, Kevin, Louise, Marnie, Nicole, Oriane, Patrick)
H2	<b>Maths</b> (Kel, Sandy, Leah, Abigail, Sam, Eliot, Aubrey, Sunny, Hoagie, Laverne, Fred, Emily, Bobette, Céline, Gerard, Sallyface, Hector, Ines, Josette Kevin)	<b>SES</b> (Queen, Ralf, Sebastien, Titouan, Ursula, Valerie, Xavier, Yvonne, Gerry, Clotho, Yorick, Gunther, Horadrick)
H3	<b>Physiques</b> (Alex, Blob, Charline, Dave, Eve, Omori, Marie, Kel, Sandy, Leah, Abigail, Sam, Eliot, Aubrey, Sunny, Hoagie, Laverne)	<b>SES</b> (Shepherd, Vagner, Cory, Beatrice, Rosa)
H4	<b>Maths</b> (Louise, Marnie, Nicole, Oriane, Patrick)	<b>SVT</b> (Alex, Blob, Charline, Dave, Eve)



## V. MiniZinc:

Python n'était pas un programme adapté aux problèmes de contraintes. Cependant, certains langages de programmation se prêtent mieux à ce problème, celui que nous avons choisi est MiniZinc.

Ce langage nous a permis de décrire directement les contraintes (ne pas mettre plusieurs examens dans une même salle, ne pas avoir plusieurs élèves dans la même salle etc...). Après avoir inséré les données, le programme range chaque élève dans une salle associée à un horaire, de sorte à ce que toutes les contraintes soient respectées.

Pour les résultats données ci-dessous, nous avons repris les mêmes examens et élèves, nous avons contraint le programme à 3 salles de 30 places et 2 créneaux horaires maximum. Voici les résultats:

```
H1= salle1 3      salle2 2      salle3 2
H2= salle1 4      salle2 1      salle3 1
[3, 2, 2, 4, 1, 1]
```

Mis sous forme de tableau

	Salle 1 (C = 30)	Salle 2 (C = 30)	Salle 3 (C = 30)
H1	Physiques	SES	SES
H2	SVT	Maths	Maths

(Aucun groupe d'élève ne fait Physiques et SES ou SVT et Maths)

## VI. Conclusion:

Malgré les diverses difficultés auxquelles nous avons été confrontés, nous sommes parvenus à une solution répondant à la consigne initiale : optimiser le temps et l'espace pour organiser la passation de plusieurs examens, tout en tenant compte à la fois de la disponibilité des salles et du nombre d'élèves concernés. Ce qui nous a posé le plus de problèmes a été l'optimisation du temps et l'espace, un défi qui s'est révélé ardu à résoudre.

### A. Sources et annexes:

<https://www.minizinc.org/doc-2.5.5/en/MiniZinc%20Handbook.pdf>