

# Examen 2 – PL - Objectifs

---

1. Formalisation mathématique
2. Résolution graphique
3. Résolution avec Pulp
4. POO et tests unitaires

# Formalisation mathématique

---

- Concernant un pb de même type vu en classe (production/recette...)
- Exercice 7 fichier Word
- Décrire les variables de décision
- Écrire la formule mathématique de la fonction objective et des contraintes

# Résolution graphique d'un PL

---

- Dessiner une zone de faisabilité, une fonction objective arbitraire, ajouter un titre, une légende et annoter des points
- Matplotlib (cours 08 - `res_graphique_demo.py`); `subplots()`, `plot()`, `set_xlim()`, `set_ylim()`, `legend()`, `title()`, `text()`, `annotate()`, `show()`

# Examen 2 – PL – Objectifs - Pulp

---

3 syntaxes pour créer 1 contrainte/fonction objective (demo1 – pulp – cours 07);

1. En créant une variable avec un constructeur,

```
c1 = LpConstraint(e=x + y,  
                 sense=LpConstraintLE,  
                 rhs=2,  
                 name='C1 créée par un constructeur')
```

2. En créant une variable avec une expression,

```
c1: LpConstraint = x + y <= 2
```

3. Sans créer de variable, en ajoutant une expression directement au problème

```
prob += x + y <= 2, "Encore c1 ajouté sans créer de variable"
```

# Examen 2 – PL – Objectifs - Pulp

---

```
1  \* Cours_07_Demo1_Pulp *\n2  Minimize\n3  La_même_fonction_objective: - 4 x + y\n4  Subject To\n5  C1_ajouté_pls_fois_au_pb_avec_une_nouvelle_étiquette: x + y <= 2\n6  C1_créée_par_un_constructeur: x + y <= 2\n7  C1_créée_par_une_expression: x + y <= 2\n8  Encore_c1_ajouté_sans_créer_de_variable: x + y <= 2\n9  Bounds\n10     x <= 3\n11     y <= 1\n12 End
```

# Pulp

---

- Désactiver/activer les traces par défaut du solveur et imprimer les résultats (valeurs optimales des variables, objectifs et contrainte « slack »)
- Interpréter la solution obtenue ; le statut du PL et son graphique (exercice 09)
- Modifier le problème pour répondre à d'autres scénarios de données (exercice 07)

# POO et tests unitaires

---

- Typage le code Pulp
- PEP8
- Pattern constant
- Séparer les données et la résolution du PL; utiliser des dictionnaires de données et `lpSum()` (Exercice 08 et 09)
- Intégration POO ; classes et objets, test unitaires (Exercice 09)