

## LIFAMI – TD : Collision et système Masses-Ressorts

*Objectifs :* Physique des ressorts et codage de structures masse-ressort

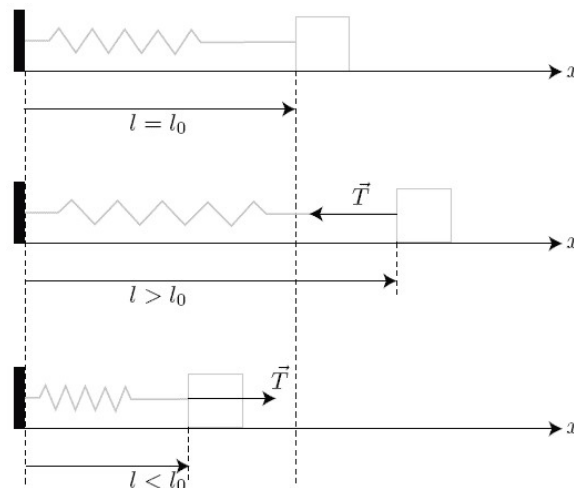
Le TD système de particules doit être entièrement terminé. S'il ne l'est pas, terminez-le.

### La balle au bond

Regardez la partie collision du TP. Les particules doivent rebondir sur les bords de la fenêtre.

### Masses-Ressorts

Nous allons aller un peu plus loin avec la manipulation de particules en mouvements en connectant deux particules ensemble avec un ressort.

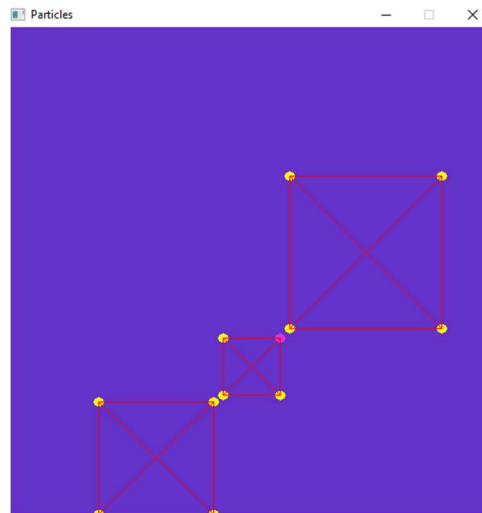


La force de rappel élastique exercée par le ressort sur la masse  $m$  est

$$\vec{T} = -k \cdot \Delta l \cdot \vec{e}_r$$

où:

- $\vec{e}_r$  est un vecteur unitaire dirigé suivant l'axe du ressort, orienté vers l'extérieur.
- $\Delta l = l - l_0$  est l'allongement du ressort en notant  $l$  la longueur du ressort, et  $l_0$  sa longueur à vide.
- $k$  est la **raideur du ressort**, intrinsèque au ressort considéré, elle caractérise la capacité du ressort à résister au mouvement de la masse  $m$ , et par conséquent à revenir à sa position d'équilibre.



1. Définissez une structure *Spring* (ressort) s'adossant à la structure *World* contenant l'ensemble des particules du monde. Vous pourrez utiliser l'indice d'une particule dans le tableau comme identifiant.
2. Ajoutez un tableau de ressorts à la structure du monde *World*
3. Modifiez la procédure *init* qui initialise le monde pour qu'elle crée également les ressorts. Nous vous conseillons d'écrire une structure *addParticle* qui renvoie l'indice de la particule ajoutée. Vous pourrez commencer par créer 2 particules reliées par un ressort, puis faire un triangle, puis un carré, etc.

```
procedure addParticle(w : donnée/Resultat World) ;
```

4. Modifiez la procédure *draw* pour qu'elle affiche les ressorts représentés par des lignes.
5. Ecrivez la procédure *computeParticleForceSpring* qui calcule les forces qu'exercent les ressorts sur toutes les particules du monde.