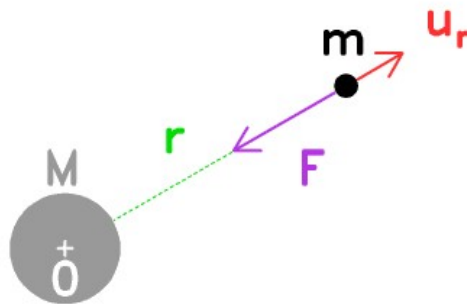


## LIFAMI – TP : Système de particules

*Objectifs :* Mécanique du point et codage en C/C++

### La base



1. Codez les questions relatives à « Et si on codait ca ? » du TD Système de particules.

Le programme principal avec *Grapic* peut ressembler à ceci :

```
int main(int , char ** )
{
    World dat;

    bool stop=false;
    winInit("Particles", DIMW, DIMW);
    backgroundColor( 100, 50, 200 );

    init(dat);

    Menu menu;
    menu_add( menu, "Init" );
    menu_add( menu, "Run");

    while( !stop )
    {
        winClear();

        draw(dat);          // dessine les particules
        update(dat);        // déplace les particules

        if (menu_select(menu)==0)
        {
            init(dat);
            menu_setSelect(menu,1);
        }
        menu_draw( menu);
        stop = winDisplay();
    }
    winQuit();
    return 0;
}
```

## La chute!

Sur terre, une approximation de la force que subit une particule soumise à l'attraction terrestre est le poids, avec  $m$  la masse et  $G$  est la constante de gravitation universelle  $G=9.81$  :

$$P = m.G$$

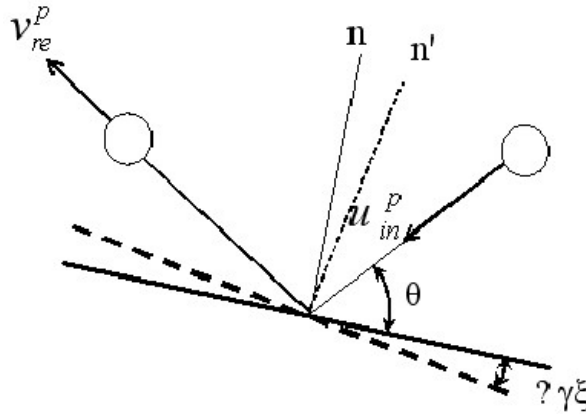
15. Ajoutez une structure *World* qui va stocker plusieurs particules dans un tableau. En complément de la structure, programmez une fonction *init*.
16. Ajoutez à votre programme cette force de gravité pour que vos particules « tombent ».

2. Pour la gestion des collisions nous allons considérer uniquement des murs horizontaux ou verticaux :  $x=0$ ,  $y=0$ ,  $x=X_{\max}$ ,  $y=Y_{\max}$

Modifiez le code de la fonction *update* pour intégrer ces collisions. Vous devriez obtenir des particules qui rebondissent dans la boîte de l'écran. Le test de collision est assez simple : `if (x<0) { ... }`

La réaction à une collision peut se faire en remettant artificiellement la particule dans le rectangle de la simulation :  $x = -x$  ;

Il faut aussi réorienter le vecteur vitesse de manière symétrique :  $v_x = -v_x$

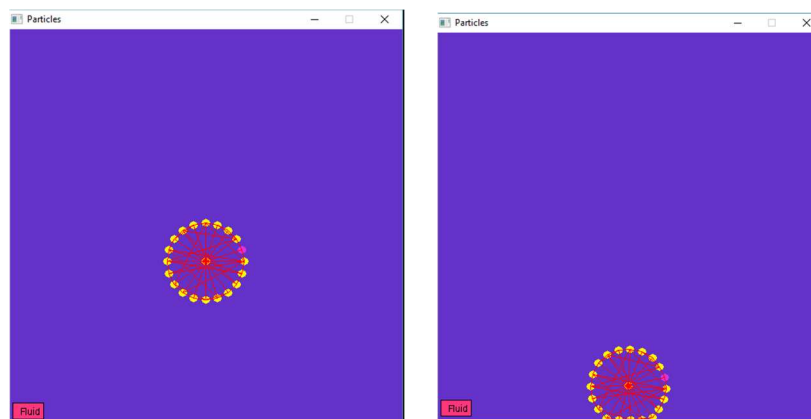


3. Après un choc une particule perd un peu de vitesse (énergie) absorbée par le mur. Ceci se fait en réduisant le vecteur vitesse :  $v = -\text{FRICTION} * v$  où *FRICTION* est une constante inférieure à 1, par exemple 0.6 .

## Rebondissons avec les masses-ressorts

La partie Loi universelle de la gravitation qui vient après peut se faire indépendamment de cette partie. Il s'agit de deux branches possible du TP.

4. Codez toutes les fonctions du TD Masses-Ressorts.
5. Vous pouvez imaginer des interactions en ajoutant des forces à certaines particules lorsqu'on appuie sur une touche.
6. Vous pouvez imaginer des formes différentes (cercles, etc.), et par exemple faire tourner une roue en appliquant les bonnes forces extérieures au système.



## Loi universelle de la gravitation (Pour les plus rapides)

Dupliquez votre programme des questions précédentes pour en garder une sauvegarde.

17. Deux particules interagissent par la force de gravité. Notons  $M$  et  $m$  leur deux masses respectives (en kg),  $r$  leur distance (en m),  $u_r$  étant le vecteur unitaire entre les deux particules. La force  $F$  que subit chaque particule est

$$\mathbf{F} = -\frac{GMm}{r^2} \mathbf{u}_r$$

Modifiez votre programme pour que les particules ne subissent plus la gravité terrestre mais plutôt pour que chaque particule se comportent comme un astre subissant la gravité de tous les autres astres et provoquant une force de gravité vers tous les autres astres.

7. En premier test vous simulerez uniquement 2 particules : un soleil au centre avec une masse assez élevée et une particule positionnée à une distance de 100 pixels.  
Quelle trajectoire allez-vous observer ?

Modifiez votre programme pour que la petite particule tourne autour de la grande.

Remarque : la particule peut sortir de l'écran.

8. Testez plusieurs configurations de particules avec différentes masses. Essayez de retrouver quelque chose qui ressemble au système solaire.

$\text{Masse\_lune} = 0,0123 * \text{Masse\_terre}$

$\text{Masse\_soleil} = 330000 * \text{Masse\_terre}$

9. Faites des tests avec des dizaines de particules.

