

## Formule de physiques

- Formule de pendule (composé et/ou simple)------(fait)
- Formule avec la gravité------(fait)
- Formules de magnétisme------(fait)
- Formule d'un plan------(fait)
- Formule indiquant de quel côté d'une droite le point est------(fait)
- Formule de distance entre deux point------(fait)
- Formule de frottement------(fait)
- Formule pour appliquer le vent------(fait)
- Formule pour les rebonds objet-Map------(fait)
- formule pour les rebonds objet-objet------(fait)

## Détail sur les procédure et fonctions de la classe Physique:

### **Vecteur3 obtenirForceNormal(Double masse, Vecteur3 Position);**

Fonction qui retourne la force normale d'un plan par rapport à un objet:

$$F_n = m * g * \cos(\theta)$$

- $m$  = Masse de l'objet
- $g$  = gravité
- $\theta$  = angle entre la surface et l'axe vertical(Donc entre obtenirNormaleSurface et l'axe!)

### **appliquerGravite(Vecteur3);**

Procédure qui applique la gravité sur un objet en l'additionnant à la composante verticale de son vecteur de déplacement.

### **appliquerVent(Vecteur3d vecteurVitesseVent, Objet& objet);**

Procédure qui applique le vent au vecteur d'un objet en addition de vecteur. Pour trouver l'accélération que le vent donne à l'objet, je vais d'abord calculer l'angle entre le vecteur du vent et la normale de chaque face de l'objet. Si l'angle est négatif, le vent applique une force sur la face. Je calcule l'aire de celle-ci et l'additionne à  $S$ , et l'angle négatif, j'additionne sa valeur absolue à  $C$  qui est le coefficient de traînée. Plus  $C$  est petit, plus l'objet est courbe alors je me fais un total de face touchée et je divise  $C$  par celui-ci pour avoir une valeur entre 0 et 1 (1 serait une face plane).

$$a = (0.5 * \text{masse volumique} * \text{vitesse vent} * C * S) / m$$

### **appliquerMagnetisme(Objet& objet, Vecteur3d positionAimant, double force);**

Procédure qui applique la force d'attraction d'un aimant sur un objet selon sa masse, sa position et la position de l'aimant. La force de l'aimant est prédéterminée à 4 tesla et la sensibilité magnétique est celle du fer.

### **appliquerFrottement(Objet);**

Applique le frottement à un objet lorsqu'il est en collision selon une constante de frottement(identique pour tous les matériaux pour l'instant, donc le bois glisse autant que la glace!) et la force normale.

### **RebondObjetCarte(gfx::Modele3D& objet1, Vecteur3d vecteurNormal);**

Procédure qui appliquera le rebonds à un objet(son vecteur) lorsqu'il entre en collision avec un autre objet fixe ou la map. La formule ressemble beaucoup à celle du worms. Par contre, pour rendre le tout, il faudra considérer la constante de restitution de l'objet en question. Voici le calcul en question :

$$(2 - \text{constante } r) * \text{vecteur vitesse de l'objet} * \text{vecteur normal du la surface}$$

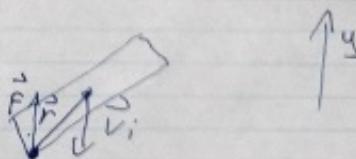
Ceci donne un scalaire qu'on multiplie au vecteur normal pour ensuite soustraire le résultat au vecteur vitesse de l'objet.

Test de calcul de rebond de fou:

$$\vec{F} = m\vec{a} = m \frac{\partial \vec{v}}{\partial t} = \frac{\partial (m\vec{v})}{\partial t} \rightarrow \vec{p}$$

$$\vec{F} = \frac{\partial \vec{p}}{\partial t}$$

$$F_{\text{avg}} = \frac{\Delta p}{\Delta t} = \frac{p_f - p_i}{\Delta t} = \frac{mv_f - mv_i}{\Delta t} \approx 0.0015$$



$$\tau = r F_{\text{avg}} \sin \theta = I \alpha_{\text{avg}} = \frac{m L^2}{12} \alpha$$

$$\alpha_{\text{avg}} = \frac{r F_{\text{avg}} \sin \theta}{I} = \frac{r \Delta p \sin \theta}{I \Delta t}$$



$$E_i = E_f$$

$$\frac{1}{2} m v_i^2 = \frac{1}{2} m v_f^2 + \frac{1}{2} I \omega^2$$

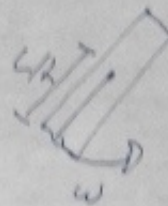
$$80\% E_i = E_f$$

$$C^2 E_i = E_f$$

Cons de E

Cons de p

$$\frac{1}{2} m v_i^2 = \frac{1}{2} m v_f^2 + \frac{1}{2} I \omega^2$$



$$\omega_f = \omega_i + \Delta t$$

$$\omega_f = \Delta \theta / \Delta t$$

$$\omega_f = \frac{r \Delta \theta \sin \theta}{I}$$

$$\omega_f = \frac{r \sin \theta}{I} m (v_f - v_i)$$

$$\frac{1}{2} m v_i^2 = \frac{1}{2} m v_f^2 + \frac{1}{2} I \left( \frac{r \sin \theta}{I} m (v_f - v_i) \right)^2$$

$$\bullet \quad \frac{1}{2} m v_i^2 = \frac{1}{2} m v_f^2 + \frac{1}{2} \frac{m^2 r^2 \sin^2 \theta}{I} (v_f^2 - 2 v_f v_i + v_i^2)$$

$$C_R = \frac{V_F}{V_i} \quad \frac{K_F}{K_i} = \frac{\frac{1}{2} m V_F^2}{\frac{1}{2} m V_i^2} = \left( \frac{V_F}{V_i} \right)^2 = C_R^2$$

$$0 = \frac{1}{2} m V_F^2 + \frac{1}{2} \frac{m^2 r^2 \sin^2 \theta}{I} V_F^2 - \frac{m^2 r^2 \sin^2 \theta}{I} V_F V_i$$

$$+ \left( \frac{1}{2} \frac{m^2 r^2 \sin^2 \theta}{I} V_i^2 - C_R^2 \frac{1}{2} m V_i^2 \right)$$

D C

$$0 = \left( \frac{m}{2} + \frac{1}{2} \frac{m^2 r^2 \sin^2 \theta}{I} \right) V_F^2$$

$$b \leftarrow \left( - \frac{m^2 r^2 \sin^2 \theta}{I} V_i \right) V_F$$

+ C

$$0 = a V_F^2 + b V_F + C$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

**RebondObjetObjet(Objet& objet1, Objet& objet2, Vecteur3d**

**vecteurNormal);**

Procédure qui appliquera le rebonds à deux objets (en mouvement) qui entrent en collision. Dépend de l'élasticité des objets.

### **Double obtenirAnglePenduleSimple(Double angleMaximal, Double frequenceAngulaire);**

Fonction qui retourne l'angle d'un pendule simple selon le temps, son angle maximal et sa fréquence angulaire.

$$\theta(t) = \theta_m \cdot \cos( \omega t )$$

- $\theta(t)$  = Angle retourné(Position angulaire du pendule)
- $\theta_m$  = angle maximal du pendule
- $\omega$  = fréquence angulaire du pendule
- $t$  = temps

### **Double distanceEntreDeuxPoints(Vecteur3 Point1, Vecteur3 Point2);**

Fonction qui retourne la distance entre deux point!!!!!!

### **Integer positionPointDroite(Vecteur3, Vecteur3, Vecteur3);**

Fonction qui retourne la position d'un point par rapport à une droite:

- -1 = Point à gauche de la droite
- 0 = Point sur la droite
- 1 = Point à droite de la droite

Les deux premiers vecteurs(point) servent à représenter la droite, le troisième est le point et le dernier vecteur est la normale du plan dans lequel le point se trouve.

### **Double obtenirEnergieCinetique(Double masse, Vecteur3 Vitesse);**

Fonction qui retourne l'énergie cinétique à partir d'une masse et d'un vecteur vitesse.

$$k = \frac{1}{2} m \cdot v^2$$

## **Détails sur les procédures et fonctions de la classe Plan:**

### **Plan(Vecteur3 p1, Vecteur3 p2, Vecteur3 p3);**

[Lien pour formule de création d'un plan](#)

Équation d'un plan:

$$ax+by+cz+d = 0$$

- $u = p1 - p2$
- $v = p1 - p3$
- $\text{normale} = u \times v$
- $d = u \cdot p1$

### **Vecteur3 intersectionPlanDroite(Droite);**

Fonction qui retourne les point d'intersection entre un plan et une droite. Retourne NULL si il n'y en a pas.

[Lien pour formules d'intersection](#)

### **void calculerPlan(Vecteur3 p1, Vecteur3 p2, Vecteur3 p3);**

Fonction qui recalcule un plan, comme dans le constructeur.

## **Détails sur les procédures et fonctions de la classe Droite:**

### **Droite(Vecteur3 p1, Vecteur3 p2);**

Calcul du vecteur directeur:

$$\text{vecteurDirecteur} = A \times B$$

## **Détails sur la classe Chrono**

Cette classe va nous donner la variable t pour nos fonctions. Par exemple, pour trouver l'angle du pendule simple, nous avons besoin du temps:

$$\theta(t) = \theta_m \cdot \cos( \omega t )$$

que nous donnera cette classe!

Le chrono sera activé au début de la partie et rafraichie à chaque frame pour avoir le temps virtuel.