1. Justifie bien que les diffèrent effets sont visibles et notre fichier de test sur les vecteurs
2. Une fonctionnalité que nous souhaitions ajouter est de gérer la génération d’image de maniéré ‘propre ‘ au lieu de tout mettre dans le main. Nous avons donc créé une classe ObjectScenario qui permet de regrouper l’ajout des objets, le calcul de l’image et son enregistrement. Cette classe pourra par exemple être utilisé pour génère plusieurs images à la suite et de rendre le code plus lisible. Ça principale utilité cependant est de grandement faciliter l’utilisation du code. Quelqu’un souhaitant s’interfacer sur notre code n’aura potentiellement que cette classe à utiliser pour générer des images.

Dans notre ‘simple’ cas pour générer plusieurs images, nous créons une classe ‘ScenarioPersoX’ héritant de ObjectScenario. Nous redéfinissons son constructeur afin d’ajouter tous les objets que nous souhaitons à l’instanciation de cette classe dérivée. Nous appelons ensuite la méthode toImage qui permet de calculer l’image liée au scenario et de l’enregistrer.

Le calcul d’un scenario personnalisé est de la forme :

ScenarioPerso(int largeur, int hauteur).toImage(char[] nom, bool afficherTexte, int profondeur);