Exercices à rendre – Feuille 2

M1-Gphy-Imagerie

1. Le code affiche un rectangle bleu décentré. Vérifiez les coordonnées du vertex buffer, et expliquez pourquoi le rectangle n’est pas au centre (l’explication est dans le cours, lisez bien la partie 5 depuis le début).

Le rectangle n’est pas centre car les coordonnées subit une translation dans le vertex shader.

1. Replacez-le au centre, sans modifier les coordonnées du vertex buffer. Donnez la ligne de code que vous avez écrite pour cela.

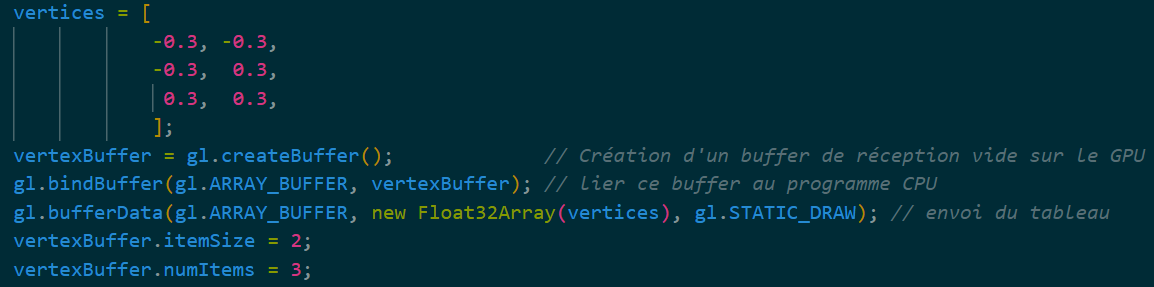
La ligne de code qui permet de replace le triangle au centre sans modifier les coordonnées du vertex buffer est : « gl.uniform1f(shaderProgram.shapeOffsetX, 0); »

1. Nous avons manipulé un vertex buffer et un vertex shader. Quelle est la différence entre ces deux éléments du code ?

La différence entre le vertex shader et le vertex buffer est que le vertex buffer géré les données et la structure de celle-ci tandis que le vertex shader géré le calcule de position à partir des données.

1. Une image contenant Bleu électrique, bleu, Graphique, drapeau

   Description générée automatiquementModifiez le code pour qu’il affiche non pas un rectangle, mais un seul triangle (rectangle). Vérifiez bien les coordonnées et faites un dessiné avant de l’exécuter, et vérifiez que l’affichage correspond à ce que vous avez dessin. Copiez le tableau de sommets dans le pdf que vous rendrez, et ajoutez une copie d’écran pour montrer le résultat.



1. Modifiez le code pour qu’il affiche maintenant un triangle isocèle. Vous expliquerez ce qu’est un triangle isocèle. Copiez le tableau de sommets dans le pdf que vous rendrez, et ajoutez une copie d’écran pour montrer le résultat.

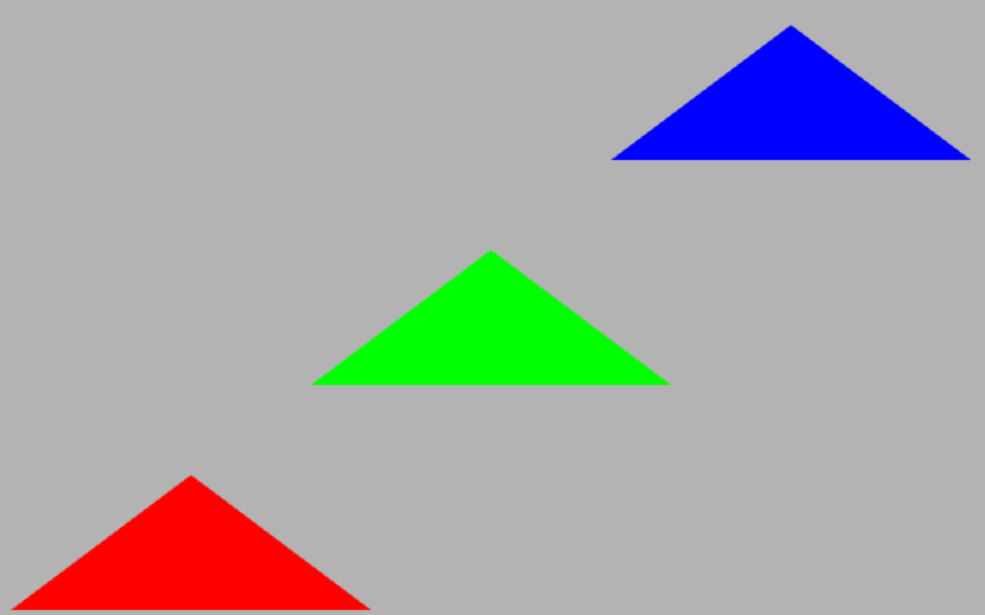
Une image contenant triangle, ligne, Bleu électrique

Description générée automatiquement

Une image contenant capture d’écran, Tracé

Description générée automatiquement

1. A partir du triangle isocèle, réalisez trois affichages (trois appels avec la fonction draw) avec la gestion des paramètres, de manière à afficher 3 fois le triangle sur la même figure, avec 3 couleurs différentes.



Une image contenant texte, capture d’écran, menu, Police

Description générée automatiquement

1. Avant d’essayer de coder la rotation, il vous faut tester le fait que cette nouvelle valeur est bien transmise correctement au vertex shader. Que proposez-vous pour réaliser ce test ?

Pour test si cette nouvelle valeur est bien transmise correctement au vertex shader, on peut faire affiche la valeur. Mais je ne suis pas sur que ce test fonctionne donc je ne sais pas.

1. La rotation doit être appliquée en premier sur les coordonnées de la variable vertexPosition. Ensuite, vous pouvez appliquer le déplacement. Pour répondre à ces questions, expliquez les difficultés que vous avez eues pour réaliser cette opération, puis donnez le code du vertex shader, et enfin ajoutez une copie d’écran avec les 3 triangles qui auront des rotations différentes et des couleurs différentes.

La difficulté rencontrée est de réussir est transformé un angle en degré Celsius en un angle en degré radiale. Pour cela j’ai rajoute une opération dans le vertex Shader.

Une image contenant Graphique, conception

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

Description générée automatiquement