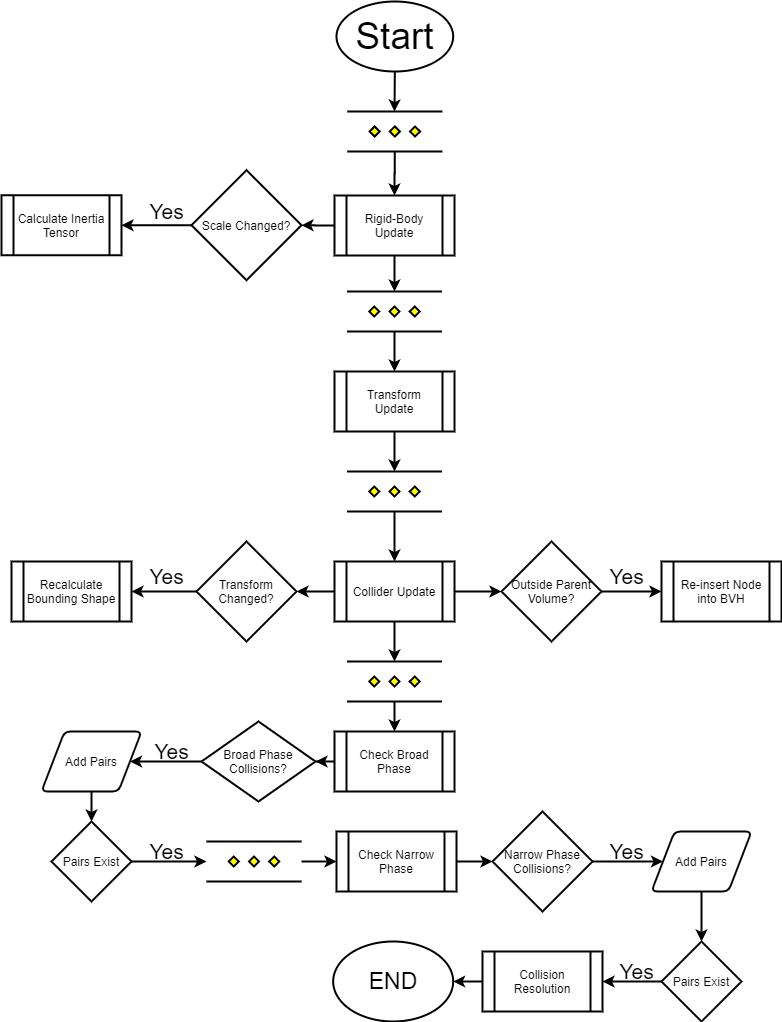
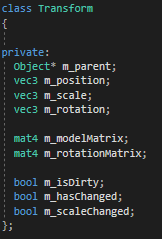
**Overview:**

The objective is creating an Engine able to stimulate rigid-bodies and detect collisions of arbitrary shapes and sizes.

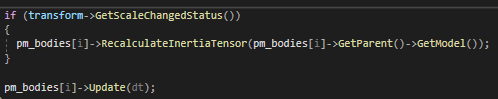
**Physics Manager Architecture: FLOW CHART**



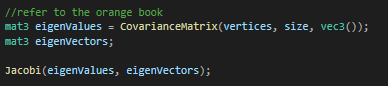
Transform:

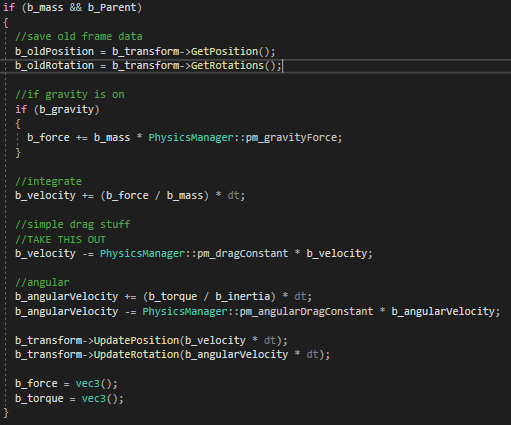
Since variables like position, rotation and scale are shared by several components they belong in a separate class accessed by other components using getters and setters (taken out of the snapshot).

Body:

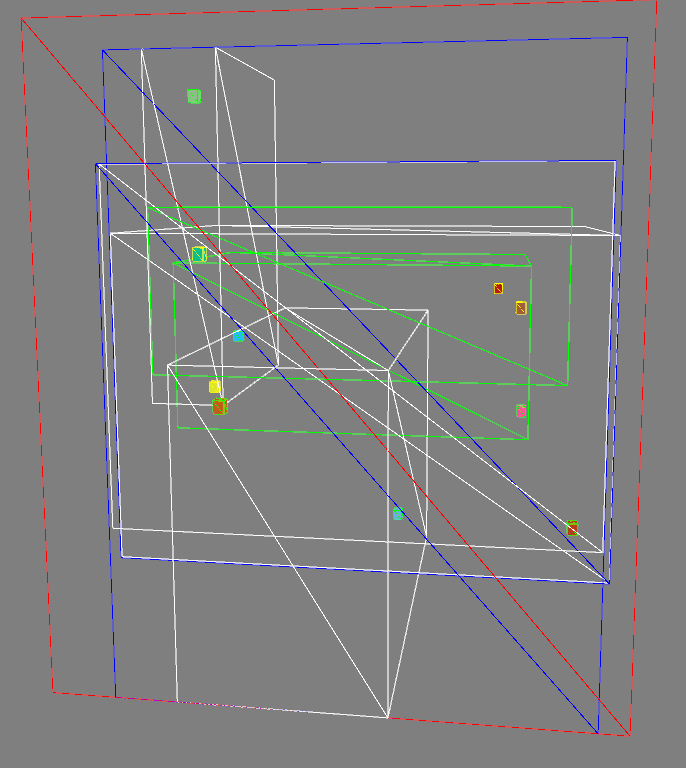


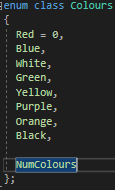
When Body is to be update, first a check is performed to see if the scale had changed in the current frame; if so, the inertia tensor is re-calculated first before updating the rigid-body.

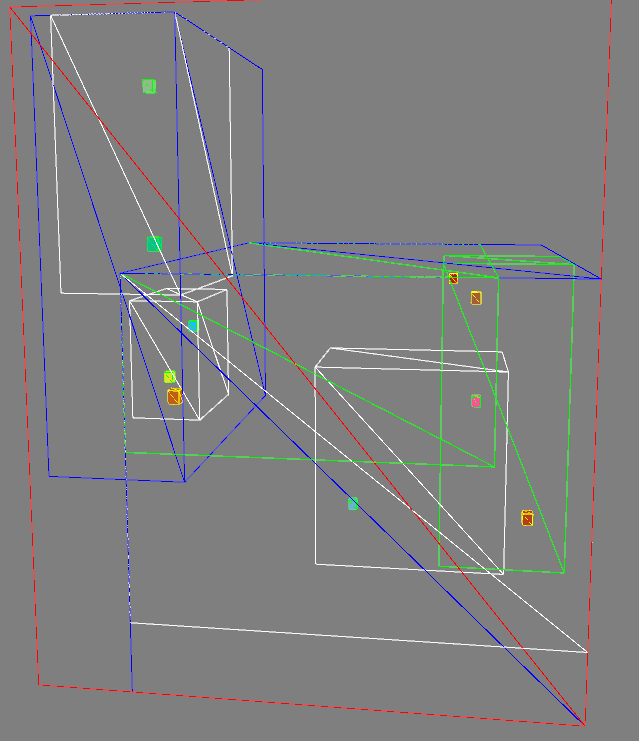
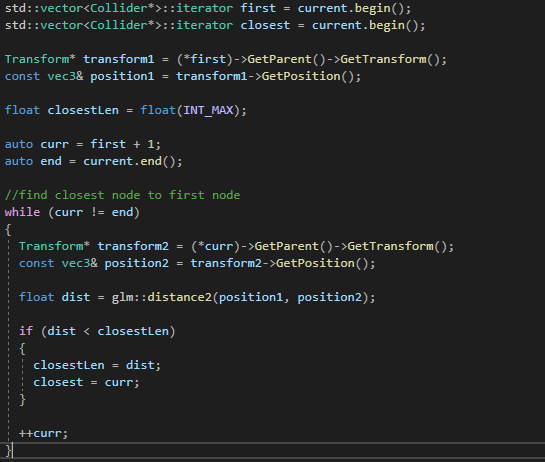
The method of calculating the inertia tensor is PCA, using co-variance matrix and Jacobi’s method. Both of which could be found in “Real-Time Collision Detection”.

The integrator is semi-implicit Euler. If the **mass of the object is 0 then it is considered a static (immovable object).**

Broad – Phase:

AABB Bounding Volume Hierarchy is used for Broad Phase Detection. The tree is always promised to be Balanced, not particularly promised to be compact.

The Levels of the tree are colored the same in a defined order.

Since the tree is not compact when creating, a separate method is implemented to rebuild the tree when necessary. The new Tree is **promised to be balanced and in compact pairs.**

The pairs are created using the closest Euclidean with the first object in the list, then the pair is removed from said list.

Narrow – Phase:

Gilbert–Johnson–Keerthi (GJK) distance algorithm will be used for narrow phase collision detection. This test will only be performed on pairs that pass the Broad phase meaning their Bounding boxes intersect. GJK was chosen for its ambiguity to object’s shape; therefore, the engine will be able to detect collision of objects regardless of their shapes. GJK was also chosen since this would be a good learning experience.

Collision Resolution:

Impulse Based collision resolution will be implemented for the extents of this course mainly for its simplicity which will allow engine to scale to a large number of objects without compromising performance.

Conclusion:

From this set of techniques, the engine should be able to scale up to a large number of objects. Detect and respond to high number of collisions arbitrary to their shapes. The main issue to look out for is to optimize GKJ as it is a very expensive computationally and can incur many iterations.

**References:**

Books:

* Ericson, Christer. *Real-Time Collision Detection*. Elsevier.