

```

#ifndef SPHFLUIDSYSTEM_H
#define SPHFLUIDSYSTEM_H

#include "particleSystem.h"
#include "ParticleGrid.h"
#include "extra.h"
#include <map>
#include <utility>
#include <vector>
#include "debugutilities.h"
#include "assert.h"

using namespace std;

class SPHFluidSystem : public ParticleSystem
{
public:
    SPHFluidSystem();
    SPHFluidSystem(int numParticles);
    virtual ~SPHFluidSystem();

    // Functions inherited from ParticleSystem
    vector<Vector3f> evalF(vector<Vector3f> state);
    void draw();
    void reinitializeSystem();

private:
    // Instance variables
    ParticleGrid particleGrid;
    //map<int, float> densityCache;
    vector<float> vecParticleDensities;
    vector<float> vecParticlePressures;

    // Physics constants
    float PARTICLE_MASS;
    float GRAVITY_CONSTANT;
    float REST_DENSITY;
    float GAS_CONSTANT;
    float VISCOSITY_CONSTANT;
    float TENSION_CONSTANT;
    float TENSION_THRESHOLD;
    float SELF_DENSITY_CONSTANT;
    float SELF_LAPLACIAN_COLOR_FIELD;

    // Helper functions
    //float calcDensity(int particleIndex, vector<int> &neighborIndexes,
vector<Vector3f> &state);
    void calculateDensitiesAndPressures(vector<Vector3f> &state);
    void initConstants();
    bool isNan(float val);
    bool isNan(Vector3f vec);

    // Different system initializations
    void testOneInitializeSystem();
    void buildTwoParticleSystemNeighbors();
    void buildTwoParticleSystemNotNeighbors();
    void build2DTestSystem();
    void buildTestSystem2();
};

#endif // SPHFLUIDSYSTEM_H

```