```
    1: #ifndef CELESTIALBODY_HPP
    2: #define CELESTIALBODY_HPP
    3:
    4: #include <iostream>
    5: #include <string>
    6: #include <vector>
    7: #include <memory>
    8:
    9: #include <SFML/Graphics.hpp>
   10:
   11: using namespace std;
   12:
   13:
   14: class CelestialBody :public sf::Drawable {
   15:
   16:     private:
   17:
   18:     double winsize;
   19:
   20:     double xpos;
   21:     double ypos;
   22:
   23:     double xvel;
   24:     double yvel;
   25:
   26:     double mass;
   27:     double radius;
   28:
   29:     double display_x;
   30:     double display_y;
   31:
   32:     string filename;
   33:
   34:     sf::Sprite sprite;
   35:     sf::Texture texture;
   36:
   37: public:
   38:     //constructors
   39:     CelestialBody();
   40:     CelestialBody(double x_pos, double y_pos, double x_vel, double y_vel,
 double m, string name, double radius, double winsize);
   41:     ~CelestialBody();
   42:
   43:     friend std::istream& operator >>(std::istream& input, CelestialBody&
ci);
   44:     friend std::ostream& operator <<(std::ostream& out, CelestialBody& co
);
   45:
   46:     virtual void draw(sf::RenderTarget& target, sf::RenderStates states)c
onst;
   47:
   48:     //accessor functions
   49:     double get_posx();
   50:     double get_posy();
   51:
   52:     double get_velx();
   53:     double get_vely();
   54:
   55:     double get_mass();
   56:     string get_filename();
   57:
   58:     //mutators
   59:     void set_x_y_pos(double x_input, double y_input);
   60:
   61:     void set_velx(double vx);
```

```
    62:     void set_vely(double vy);
    63:
    64:     void set_radius(double radius);
    65:     void set_window(double size);
    66:
    67:     void set_position();
    68:
    69: };
    70:
    71: class Universe : public sf::Drawable {
    72:
    73:     public:
    74:
    75:     Universe(); // basic constructor
    76:     Universe(double radius, int window, int num_of_planets, istream &in);

    77:
    78:     virtual void draw(sf::RenderTarget& target, sf::RenderStates states)c
onst;
    79:
    80:     friend ostream &operator <<(std::ostream& out, const Universe& co);
    81:
    82:     void step(double seconds);
    83:
    84:     double get_r();
    85:     int get_numPlanets();
    86:
    87:     void printInfo();
    88:
    89: private:
    90:
    91:     double r;
    92:
    93:     int numplanets;
    94:     int winsize;
    95:
    96:     vector <std::unique_ptr <CelestialBody>> planets;
    97:
    98: };
    99:
   100: #endif
```