```cpp
1  #pragma once
2
3  #include <iostream>
4  #include <vector>
5  using namespace std;
6
7  #include <GL/glew.h>
8  #include <GLFW/glfw3.h>
9  #include <glm/glm.hpp>
10
11  class Shapes {
12
13  public:
14      Shapes();
15      ~Shapes();
16
17      void Load();
18      void Draw();
19      void  checkErrorShader(GLuint shader);
20
21      vector<GLfloat> vertexPositions;
22
23      GLuint          program;
24      GLuint          vao;
25      GLuint          buffer;
26      GLint           mv_location;
27      GLint           proj_location;
28      GLint           color_location;
29      glm::mat4       proj_matrix = glm::mat4(1.0f);
30      glm::mat4       mv_matrix = glm::mat4(1.0f);
31
32      glm::vec4 fillColor = glm::vec4(1.0, 0.0, 0.0, 1.0);
33      glm::vec4 lineColor = glm::vec4(0.0, 0.0, 0.0, 1.0);
34      float lineWidth = 2.0f;
35
36  protected:
37      string rawData;         // Import obj file from Blender (note: no
           textures or UVs).
38      void LoadObj();
39  };
40
41  class Cube : public Shapes {
42  public:
43      Cube();
44      ~Cube();
45  };
46
47  class Sphere : public Shapes {
48  public:
49      Sphere();
50      ~Sphere();
51  };
52
```

```cpp
class Arrow : public Shapes {
public:
    Arrow();
    ~Arrow();
};
```