


《京程一灯》精英班第九讲试卷

姓名：

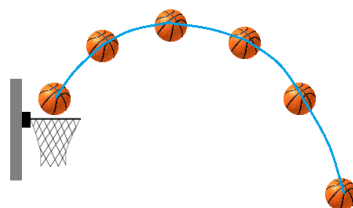
请按要求作答，并工整书写试卷。

- 1.斜抛运动是指物体有向上的初速度仅受重力作用的运动，请分析此运动，写出速度与位移的关系。

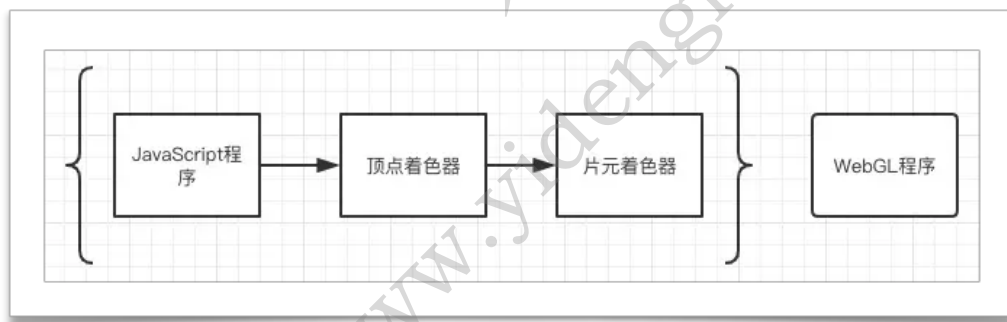
 本题考点分为如下：

速度 水平方向：匀速运动 $V_x = V_x$;
 竖直方向：竖直上抛运动 $V_y = V_y + gt$;
 位移 水平方向： $X_x = V_x * t$;
 竖直方向： $X_y = V_y * t + \frac{1}{2}gt^2$;

竖直方向：竖直上抛运动 $V_y = V_y + gt$;



- 2.请简述一下WebGL怎么把数学模型转成图片(8分)

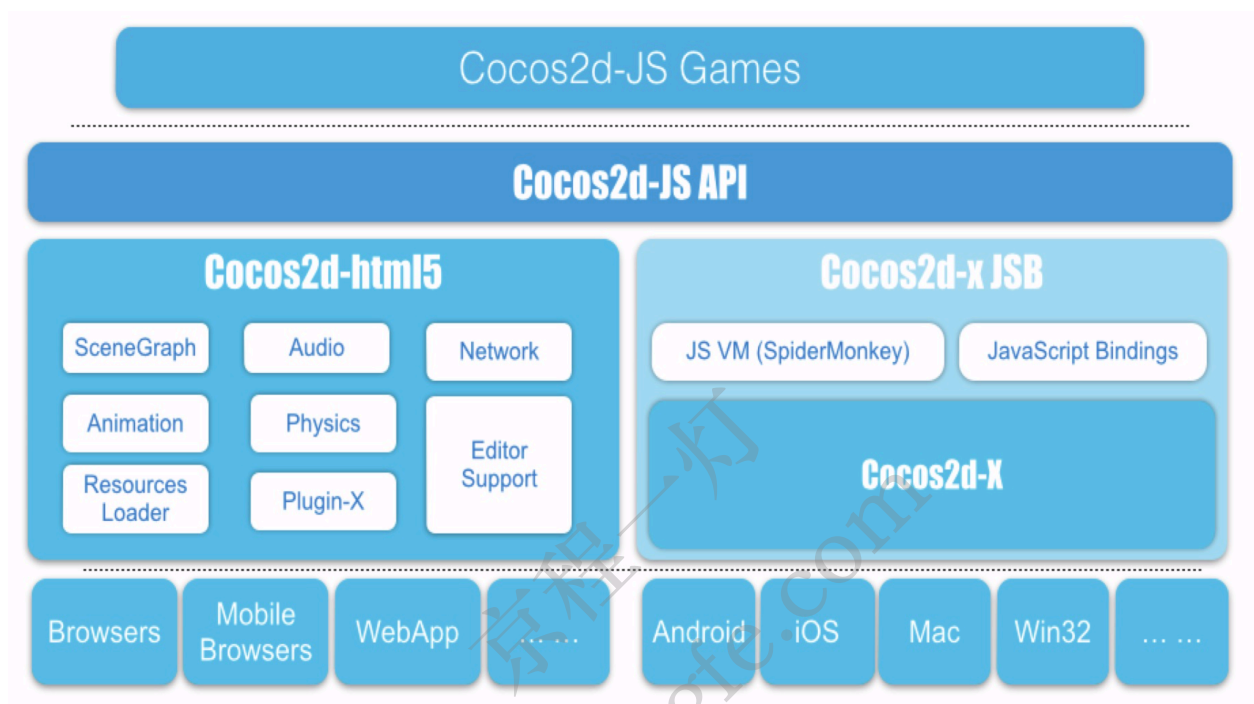


在构建 3D 物体时通过顶点组成三角形网格，但这些三角形网格都是矢量图形，最终在屏幕上显示时还是需要转化为像素图形，这种转化过程被称为光栅化，是计算机图形学的关键技术之一。

WebGL 程序和普通的 JavaScript 程序不一样，WebGL 程序除了 JavaScript 部分之外，还包含两个着色器程序，分别是顶点着色器和片元着色器；在 ThreeJS 基础中曾提到过描述一个 3D 物体，需要轮廓、材质和纹理三个要素；那么可以简单的理解为顶点着色器是用来处理物体轮廓的程序，片元着色器是用来处理物体材质和纹理的程序。

- 3.请详述Cocos2d-js知识体系。(12分)

- 答：



- 4.请描述你对物理引擎的了解，包括Chipmunk和Box2d (20分)

- 答：这个真的要好好看看咱们讲的物理引擎那节课，然后尝试发布写微信小程序对你的职业生涯是绝对有帮助的。

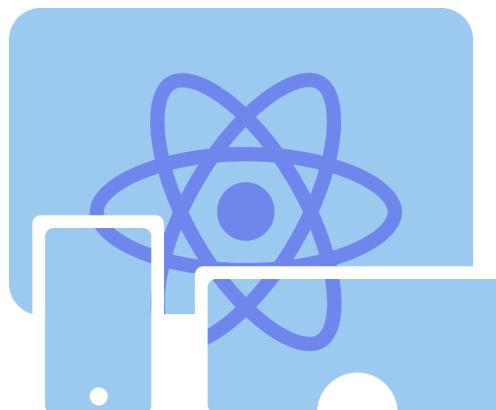
- 6.如何利用你现有的前端技术体系开发一款VR应用。(10分)

- 答：

AR.js + React-VR + React-AR

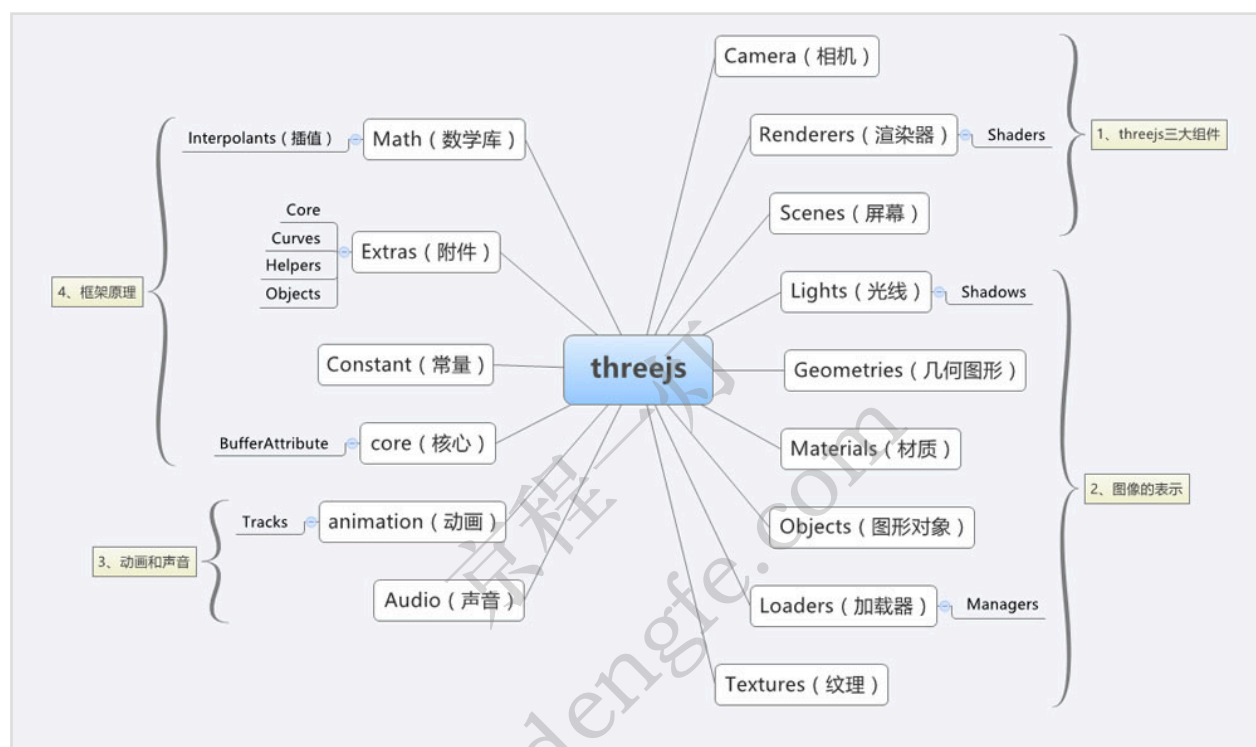
<https://github.com/jeromeetienne/AR.js>

<https://github.com/facebook/react-360>



- 7.请描述你理解的WebGL知识，并描述利用Three实现基本3D空间的必备模块？(15分)

- 答：



- 8.请利用Canvas + JavaScript 制作图片粒子效果。（5分）

- 答：

- 🍄 本题考点分为如下：

首先用Canvas的getImageData接口，获取图像的像素信息。然后了解像素区域数据的排布说明，以上获取的图片数据像素信息(imageData对象中的data属性)为RGBA整型的一维数组数据。一个像素是有4个值 (R,G,B,A) 组成的。也就是说，数组信息每四个为一个像素点。

因此，有以下规则，第一个像素信息为：RGBA(data[0],data[1],data[2],data[3])。第二个像素信息为：RGBA(data[4],data[5],data[6],data[7])第N个像素信息为: RGBA(data[(n-1)*4],data[(n-1)*4+1],data[(n-1)*4+2],data[(n-1)*4+3]) 另外，像素区域既然是一个区域，它是有宽和高的。上面的推算公式适合单独一行使用定位一个像素点。所以计算像素点时要考虑到在整个图像区域内定位：

如果图像的宽和高都为200，如果按照每一个像素为一行一列时。则该图像共有200行，200列。所以要取得 i 行第 j 列的像素初始位置信息为：

```
var pos = [(i-1)*200+(j-1)]*4;
```

其中，公式中的 i 表示行数，j 表示列数。200为图像的宽度。

- 具体见群内代码。

- 9.ThreeJS的光源有几种呢，每种光源都能表示什么样的光呢？(10分)

- 答：

- 1.环境光源没有位置概念，会将颜色应用到场景的每一个物体上，主要作用是弱化阴影，给场景添加颜色
- 2.点光源类似于照明弹，朝所有的方向发光，因此不产生阴影
- 3.聚灯光源类似于手电筒，形成锥形的光束，随着距离的增加而变弱，可以设定生成阴影
- 4.方向光光源类似于太阳，从很远的地方发出的平行光束，距离越远，衰减的越多
- 5.想要一个自然的室外效果，除了添加环境光弱化阴影，添加聚光灯为场景增加光线，还需要使用半球光光源将天空和空气以及地面的散射计算进去，使得更自然，更真实
- 6.平面光光源定义了一个发光的发光体，需要使用webgl的延迟渲染机制
- 7.眩光效果，在有太阳的时候使用眩光光源，会使得场景更真实

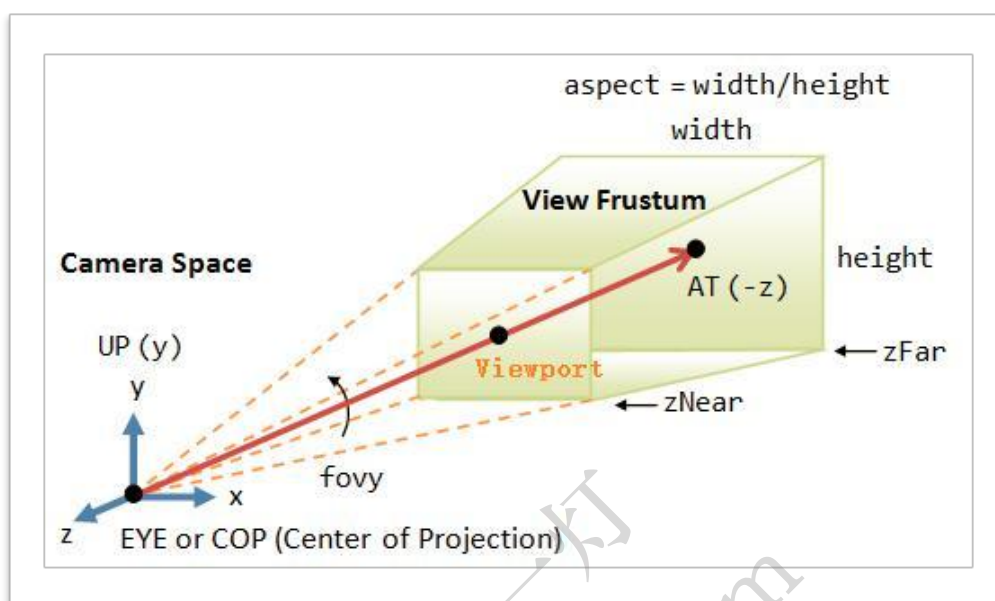
- 10.ThreeJS的材质有几种呢，每种材质都能表示什么样的效果呢？(10分)

- 答：

- 1.MeshBaseMaterial(网格基础材质)是一种非常简单材质，这种材质不会考虑光照的影响。使用这种材质网格渲染成简单的平面多边形，并且可以显示几何体的线框。
- 2.MeshDepthMaterial(网格深度材质)使用该材质的物体的外观不是由某个材质属性决定的，而是由物体到相机的距离决定的，离相机越近越亮，离相机越远越暗。该材质的属性很少，没有设置物体颜色的属性。如果想改变物体的颜色，就需要创建多材质的物体。
- 3.MeshNormalMaterial(网格法向材质)通过法向量来映射RGB颜色。每个法向量不同的面都会赋予不同的颜色。
- 4.MeshFaceMaterial(网格面材质)可以为几何体每一个面指定不同的材质。比如一个立方体有六个面你可以为每个面指定一个材质。
- 5.MeshLambertMaterial(网格朗伯材质)用于创建看上去暗淡的、不光亮的表面，可以对光源产生阴影的效果。
- 6.MeshPhongMaterial(网格phong式材质)用于创建光亮表面的材质。可以产生阴影的效果。
- 7.ShaderMaterial(着色器材质)该材质是最复杂的一种材质，可以使用自己定制的着色器。

11.ThreeJS透视效果的摄像你是怎么理解的呢？(10分)

- 答：



12.你有过3D项目开发的经验么，请问如何优化大型的3D项目？(10分)

- 答：

我们在实战的阶段除了使用draco（涵盖了WebAssembly）算法压缩模型，还使用了底图没有直接绘制进webgl👊，这里也可以开启多线程的方式进行复杂的运算，我们也可以借助一些性能监控工具进行实时的调试。核心数据一定是FPS~

前端路漫漫，这是最好的结束，也是全新的开始。京程一灯永远是您的后盾，无论何时需要帮助，我们永远都在，无论你在哪，如果需要任何帮助请随时联系我们，祝好~

