



**认证**

**专家**

**技术美术师：  
着色器和特效**

# 考试目标

认证专家  
技术美术师：  
着色器和特效

# 职责

着色器和特效技术美术师专注于实现游戏背后的视觉意图。在着色器和特效方面具备出色技能和能力的美术师常常与其他技术美术师和粒子特效美术师携手合作，以准备资源或改进所准备的资源。着色器和特效美术师负责实现游戏的外观、风格、主题和美感。

具备这些核心技能的美术师要实现实时照明和预设照明，创建和自定义着色器和渲染系统，并创建与其他资源互动的粒子系统和特效。

## 此角色的职位头衔

- 着色器编写师
- 照明师
- 特效美术师

# 资质要求

此专家认证适合在此领域工作多年、积累了深厚实践经验的专业人员。相关资质要求包括：

- 具备视频游戏开发工作室从业经验，至少参与制作过两款已上市的游戏
- 在基于物理的照明技巧和工作流方面具备深厚知识
- 在为基于物理的渲染管线制作材质方面有深入理解
- 对颜色校正和后期效果有深入理解
- 具备深厚的摄影理论知识
- 具备使用 HLSL、CgFX 或其他着色语言编写着色器的经验
- 具备使用 C++、C# 或 Unityscript 等语言编写脚本/代码的知识
- 在粒子系统、动态模拟和 Alembic 等互换格式方面具备深厚知识
- 熟练掌握 Adobe Creative Suite、Substance Designer、Substance Painter、Quixel Suite 等资源制作工具
- 对 2D 和 3D 数学理论有深入理解

# 核心技能

“着色器和特效技术美术师”认证会考查报考者是否具备所需的技能，从而有效地实现游戏的外观、风格、主题和美感。只有在下列领域具备丰富经验的报考者才能通过认证。

## 原型设计

- 制作并评估材质和着色器原型

## 着色器和材质

- 构建并测试自定义着色器，以实现下列目标：
  - 模拟各种现象
  - 根据游戏事件进行动态变更
  - 扩展标准着色器的功能，以支持视效开发 workflow
  - 实现自定义照明模型和非拟真 (NPR) 外观
- 设计、构建并实现可根据场景设计和输入而变化的程序化材质和材质效果
- 使用 ShaderGUI 实现自定义材质 UI
- 使用 OnInspectorGUI() 创建自定义检视面板 (Inspector)
- 实现后期效果 (如景深、颜色校正、泛光、屏幕空间反射、动态模糊和雾效)，以匹配 GDD 中提及的特定电影摄影技巧
- 为渲染材质的使用编写脚本，以管理实时反射效果

## 渲染和照明

- 了解不同类型的光源及其对性能的影响
- 了解不同类型的阴影及其对性能的影响
- 了解正向和延迟渲染路径之间的差异
- 确定各个平台对渲染 API 的要求和其局限性
- 使用 Unity API、命令缓冲区和图形库来调整和扩展渲染管线

## 粒子系统

- 使用多个粒子系统来模拟大气现象
- 实现火焰、爆炸、烟雾和水体等典型游戏效果
- 创建复杂粒子效果，包括具有子喷射器 (Sub-Emitters) 的粒子系统以及线条和轨迹渲染器
- 为粒子系统事件编写脚本，以便在游戏过程中根据玩家行为、NPC 行为及其他运行时事件触发此类事件
- 导入和渲染外部生成的模拟数据
- 动态评估碰撞体和变换数据，以实现与粒子系统的交互

## 性能和优化

- 了解目标平台的规格和限制
- 优化着色器、粒子系统、后期效果、照明、雾效、阴影等，使其能够在目标平台上流畅运行
- 了解该在何时使用各种优化技巧并在需要时解决出现的问题（公告板 [Billboard]、Alpha 排序问题、绘制调用、填充率问题、与 CPU/GPU 相关的性能问题）
- 使用帧调试器和各平台专用的帧捕捉工具来分析和评估渲染问题

# 认证 考试主题

---

## 工具和流程管理

- 资源定制化
  - 通过定制化的工具和编辑器改善 workflow
- 

## 渲染

- 渲染管线
  - 后期处理效果
  - Unity 摄像机
- 

## 着色器

- 着色器构建、原型设计和自定义
  - 渲染设置着色器知识
  - 与着色器相关的脚本知识
- 

## 粒子和效果

- 粒子系统自定义和扩展
  - 特效开发技巧
- 

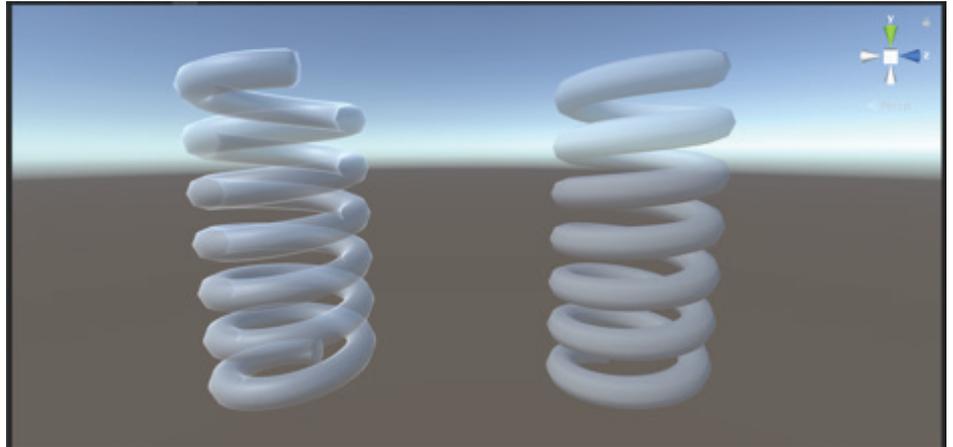
## 性能

- 场景优化

# 示例问题

## 问题 1

请参见图示：



“渲染模式”（Rendering Mode）设置为“透明”（Transparent）的标准着色器已应用于单个网格。一些面重叠并且使用了朝向相同方向的法线。左边的模型渲染不正确。概念图要求到达右侧图像中的效果。

**技术美术师应该如何达到这个效果？**

- A** 使用自定义剔除（Culling）模式。
- B** 使用自定义渲染队列。
- C** 编写自定义多通道着色器。
- D** 使用 GrabPass 编写自定义着色器。

## 问题 2

使用支持 OpenGL ES 3.0 的最小平台的移动战略游戏，其游戏设计文档（GDD）定义了一个在运行时按照程序放置产生的金矿的平面地图。金矿一部分在地下，一部分在地面。

**显示金矿入口最有效的方法是什么？**

- A** 在运行时按程序方式生成地图网格。
- B** 在地图上使用计算着色器（Compute Shader）。
- C** 在地图上使用视差着色器（Parallax Shader）。
- D** 使用带模板遮罩的着色器。

# 问题 3

第一人称冒险游戏将场景设置在具有动态照明的范围受限的多山草地上。技术美术师需要在程序性的天空颜色上添加一个程序性的云系统，并在远处添加连绵不断、无法企及的山脉。场景中的山地必须受动态光雾影响。

## 技术美术师应该如何避免山脉与云之间的相互干扰？

- A** 使用 `CameraEvent.BeforeSkybox` 添加山脉。
- B** 将山脉作为低分辨率几何体和细节等级（LOD）添加到场景中。
- C** 使用 `CameraEvent.AfterEverything` 添加山脉。
- D** 将山脉作为“世界空间用户界面图像”（World Space UI Images）添加，每个图层位于各自的“画布”（Canvas）上。

# 问题 4

技术美术师需要模仿地球上的降雨。数据是从天气服务器实时检索的。天气数据显示为彩色降雨图像，图像与全球常规漫反射纹理一致。现有一个函数可以用来抽取地球上的某个位置来获得降雨强度。游戏设计文档（GDD）要求降雨要呈现雨滴降落形式，类似于电视上的新闻天气图。

**技术美术师在运行期间将降雨分布和强度数据传递给发射器（Emitter）的最有效方法是什么？**

**A**

- 1.使用 Shape 模块设置为从全球网格的缩放实例向下发射。
- 2.将降雨纹理应用到粒子材料上使无雨地区不可见或被省略。

**B**

- 1.在脚本更新中，获取全球可见区域的随机点并检索降雨数据。
- 2.将这些数据应用于 emit 函数中的样式粒子。

**C**

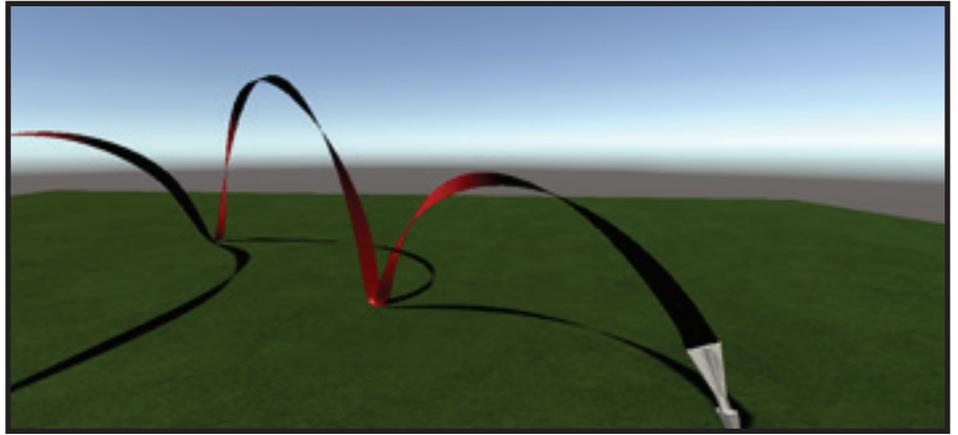
- 1.使用 Shape 模块设置为从球体向下发射。
- 2.将降雨纹理应用到粒子材料上使无雨地区不可见或被省略。

**D**

- 1.在脚本更新中，获取球形发射器的活动粒子列表。
- 2.利用它们的位置检索降雨数据，去除无雨区域的粒子，并更新有效雨粒的强度。

# 问题 5

请参见图示：



一个游戏项目包括一个玩家控制的武器，该武器可发射弹跳发射物，这些发射物在轴上旋转并留下一条显示拖痕的扭曲的色带。如图所示，色带必须对场景照明正确反应。

**创建拖痕效果的最有效的解决方案是什么？**

- A** 在与发射物关联的 GameObject 上创建拖痕渲染器 (Trail Renderer) 。
- B** 在与发射物关联的 GameObject 上创建脚本驱动的线渲染器 (Line Renderer) 。
- C** 使用 Mesh 类创建一个脚本来实现发射物效果。
- D** 创建一个脚本来驱动使用 SkinnedMeshRenderer 装配的色带模型。

---

正确答案：C、D、A、B、C