Computer Graphics II – Shadow Mapping (Questions)

Kai Lawonn

Possible Questions

What are the two passes of shadow mapping?

Possible Questions

What are the two passes of shadow mapping?

- 1st render the depth map
- 2nd render the scene as normal and use the generated depth map to calculate whether fragments are in shadow

Possible Questions

How is the `lightSpaceMatrix` defined?

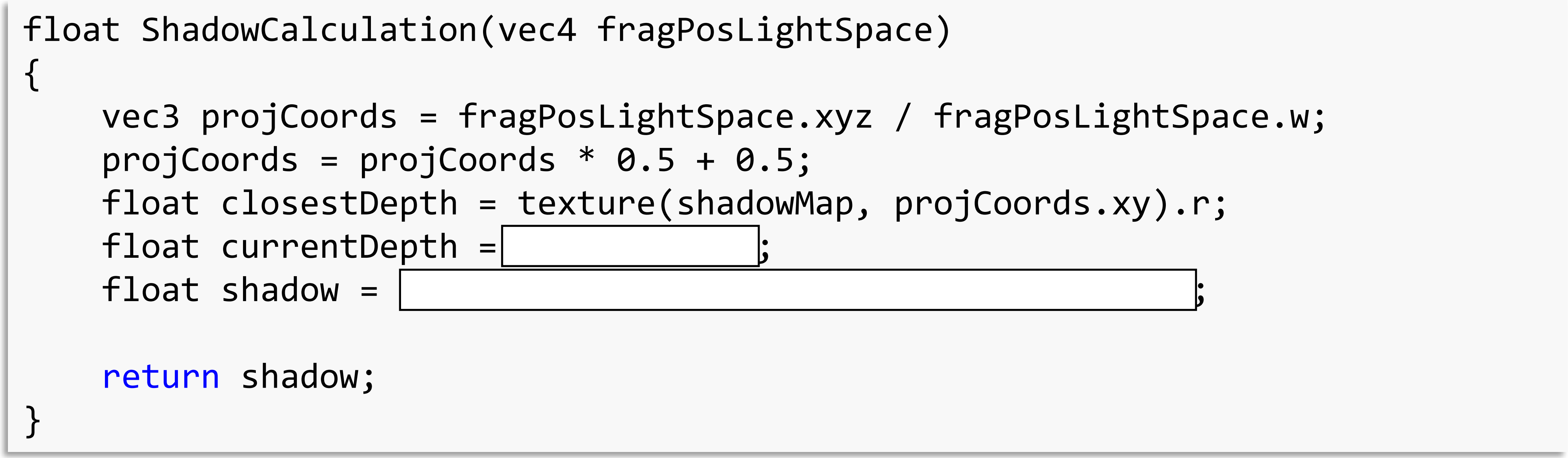
Possible Questions

How is the `lightSpaceMatrix` defined?

```
//For example:
lightProjection = glm::ortho(-10.0f, 10.0f, -10.0f, 10.0f, 1.0f, 10.0f);
lightView = glm::lookAt(lightPos, glm::vec3(0.0f), glm::vec3(0.0, 1.0, 0.0));
lightSpaceMatrix = lightProjection * lightView;
```

Possible Questions

- Complete the code:

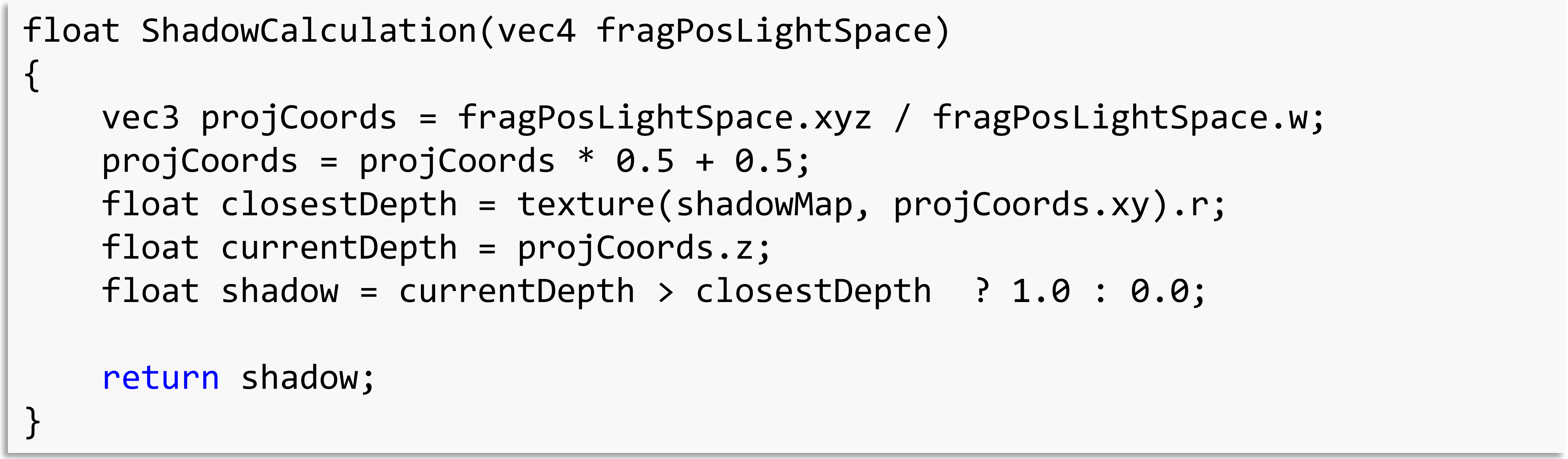


```
float ShadowCalculation(vec4 fragPosLightSpace)
{
    vec3 projCoords = fragPosLightSpace.xyz / fragPosLightSpace.w;
    projCoords = projCoords * 0.5 + 0.5;
    float closestDepth = texture(shadowMap, projCoords.xy).r;
    float currentDepth =□;
    float shadow = □;

    return shadow;
}
```

Possible Questions

- Complete the code:



```
float ShadowCalculation(vec4 fragPosLightSpace)
{
    vec3 projCoords = fragPosLightSpace.xyz / fragPosLightSpace.w;
    projCoords = projCoords * 0.5 + 0.5;
    float closestDepth = texture(shadowMap, projCoords.xy).r;
    float currentDepth = projCoords.z;
    float shadow = currentDepth > closestDepth  ? 1.0 : 0.0;

    return shadow;
}
```

Possible Questions

What is the idea of percentage-closer filtering (PCF)?

Possible Questions

What is the idea of percentage-closer filtering (PCF)?

- Produce softer shadows
- Idea is to sample more than once from the depth map, each time with slightly different texture coordinates
- For each individual sample we check whether it is in shadow or not
- All the sub-results are then combined and averaged and we get a nice soft looking shadow