CodiMD © ©	CHANGED 4 MONTHS AGO GAMES101-RUST- This is the lab for JOHN Class 2 The project framework: 作业相 Teaching Assistants: Kr.Cen、Y GAMES101-RUST-Intro	Intro 2023 Programming Practices 实链接	◆新建 ピ发表 菜单▼ W 1 ONLINE ***LOCKED GAMES101-RUST-Intro INTRO LAB1: Rotate a triangle LAB2: Rasterize triangles LAB3: Pipeline and Shading FRAMEWORK DEADLINE REFERENCE Expand all Back to top Go to bottom
	 INTRO LAB1: Rotate a triangle TASK1: Implement a rota TASK2: Let the triangle LAB2: Rasterize triangles LAB3: Pipeline and Shading FRAMEWORK DEADLINE REFERENCE INTRO 日程 		Go to bottom
	 H程 Week 1: 介绍项目与前期准备(Task 0)、熟悉 Rust 语法(Task 1)、学习完成教程 book 1(Task 2) Week 2: 学习完成教程 book 2(Task 3)和多线程优化内容(Task 4) Week 3~4: GAMES101-RUST 项目 Week 4: 结课展示 & code review 分数: Raytracer: 55分基础分,5分Bonus GAMES101-RUST: 45分基础分,5分Bonus 结课展示 & Presentation: 5分Bonus 分数占比 		程
	Raytracer Task 0 Raytracer Task 1 Raytracer Task 2 Raytracer Task 3 Raytracer Task 4 Rest of Raytracer GAMES101 LAB 结课展示 & Presentation	5% 5% 15% 20% 10% ≤ 5% (As bonus) 45% + 5% bonus 5% (As bonus)	
	TASK1: Implement a rotated in Given 3 points in 3D $v_0(2.0, 0.0, -2)$ convert these coordinates into screet In our code framework, we've alread construct transformational matrix. What you need to do? • Use model, view, projection the argument of model is rotated in the argument	plified version of rasterizer renderer based on CPU. matrix and a perspective projection matrix 0 , $v_1(0.0, 2.0, -2.0)$, $v_2(-2.0, 0.0, -2.0)$, you show the coordinates, and visualize the corresponding trianally offered draw_triangle function, so you just need the function to visualize the triangle (exactly 3 lines), when the content of the corresponding trianally of the corresponding trianall	uld ngle. d to where
	 Hints: (this may help) get_model_matrix(rotation_are 在此函数中,你只需要实现三维中。get_view_matrix(eye_pos: V36 随之旋转) get_projection_matrix(eye_for f64):使用给定的参数逐个元素量 	ngle: f64): 逐个元素地构建模型变换矩阵并返回该矩绕 z 轴旋转的变换矩阵,而不用处理平移与缩放。d): 根据摄像机的角度旋转物体(理解为摄像机摆正,sov: f64, aspect_ratio: f64, z_near: f64, z_far也构建透视投影矩阵并返回该矩阵。 image	物体:
	TASK2: Let the triangle rotate In TASK1, we've made the triangle ro	e around arbitrary axis otate around z-axis by a mvp matrix (i.e., projection *	view
	* model). Now, you need to do some any axis as you set it by yourself. What you need to do? • Use Rodrigues' Rotation Form Hint: (this may help) • 你可能需要额外写一些函数来实现 • 比如,你可以在光栅器内添加一个 draw_triangle 函数)将原来的m get_rotation(axis: Vector3<(这个参数要通过 std::io 在初始代 arbitrary_rotation ,当没有担 (即参数为z-axis, 0度)。	thing challenging. Namely, let the triangle rotate around arbitrary axis.	und (即 旋转
	 pixels with color, this is actually rast a function rasterize_triangle. What you need to do? Rasterize the triangle by function inside rasterize. Hint: (this may help) This function is usually implement 1. Create the AABB 2. Pass through all pixels in AA center of the pixel were use 	y 3 lines. Now, we want to color the triangle, i.e., set the triangles (三角形栅格化). You need to impler the triangles (三角形栅格化). You need to impler the normal natural nat	ment at the
	(you may need a inside_tr3. If inside_triangle is true its location to the correspondence of the camera, reset	riangle function with a bool return value) , then compare the interpolated depth value (插值深度 nding value in the buffer (深度缓冲). If the current poi pixel color and update depth buffer. get a picture like (We've set the triangles properly, in	度) at int is
	antialiasing. What you need to do? 用 super-sampling 处理 Anti-all 缘会有锯齿感。我们可以用 super 2 采样,并比较前后的结果 (这里的点有,对于像素内的每一个样组护一个 sample list。最后,如的黑边。 ● 要求1: 你需要用至少两种方法处理	nd some alias (黑边,即锯齿). In this part, you need to liasing: 你可能会注意到,当我们放大图像时,图像 er-sampling 来解决这个问题,即对每个像素进行 2型并不需要考虑像素与像素间的样本复用)。需要注意本都需要维护它自己的深度值,即每一个像素都需要果你实现正确的话,你得到的三角形不应该有不正常	A边 2 * 意 要 常
	 要求1: 你需要用至少两种方法处理 Aliasing, 是 SSAA 的优化),对于 以自己设计一种采样方法进行优化 要求2: 你需要给出抗锯齿优化前,对于 MSAA 方法,你应该可以得到 要求3: 你需要给出抗锯齿优化前,多少 negative effects。RUST 关于 	第二种方法,你可以去网上自行查阅抗锯齿优化的方法, ;。	也可
	多少 negative effects。RUST 关于来获取当前时间。 • 要求4: 如 tutorial 中所述,你需要解。 LAB3: Pipeline and Shad Your tasks: 1. Modify function rasterize_tri Job 2 here to realize the interpo 2. Modify function get_projection in previous labs. After this step, 3. Modify function phong_fragmen	于这方面的 API,你可以使用 std::time::Instant::n 要在 report 中给出你对光栅器产生走样(锯齿)的本质原因	和ow() 因的理 ar to hed it mage.
	implemented Blinn-Phong, the Texture Shading Fragment.5. Modify function bump_fragment implemented Blinn-Phong, car Bump mapping.6. Modify function displacement_	ent_shader: Based on the progress that you've already notice in view texture color as k_d in the formulas, and imple _shader: Based on the progress that you've already refully read the comments in this function and implement_shader: Based on the progress that you've ping, then implement displacement mapping.	ment
	GAMES101 The order of the diagram above 1. Normal shader 2. Blinn-Phong shader 3. Texture shader	GAMES 101 GAMES 101 GAMES 101	
	3.Texture shader 4.Bump shader 5.Displacement shader 7. (Optional)Have fun! Try to use of 找到其他可用的.obj 文件,抗模型也应该包含 Vertex Nore You can even make an obj ff apps! (If you do so, you will shader) as a usually gives pretty good results interpolation to do texture samp 使用双线性插值进行纹理采标 getColorBilinear(&self, 过 fragment shader 调用它实现完 Bilinear Interpolation 它。为了使双线性插值的效果Report 中同时提交纹理插值 9. After finishing the lab, you shou FRAMEWORK	file on you own. Use Blender , Meshlab or other I get extra credits) r Interpolation (双线性纹理插值). Bilinear interpolation s at reasonable costs. In this lab, you should use bilin	n near 通 要在
	并在 utils.rs 中实现 获取变换矩阵 的功rasterizer.rs 文件中定义了 Rasterizer.rs 文件中定义和文的文件中定义和文件中定义和文件中定义和文件中定义和文的文件中定义和文件中定义和文件中定义和文文和文件中定义和文件中定义和文件中定义和文文和文件中定义和文件中定	能。对于 LAB2,你需要利用 depth_buf 来实现功能。erizer 类,部分解释如下: 结需要在屏幕上绘制的颜色数据。 atrix4 <f64>):将内部的模型矩阵作为参数传递给光栅化器 (4<f64>):将视图变换矩阵设为内部视图矩阵。 etion: Matrix4<f64>):将内部的投影矩阵设为给定矩阵 dia. tor3<f64>,color: &Vector3<f64>):将屏幕像素点(x,y) n帧缓冲区位置。 我们首先定义了光栅化器类的实例,然后设置了其必要的 可便编码三角形(一般来说,不建议修改)。在主函数上, 可即可jection矩阵的函数,每一个函数都会返回相应的 set_model(),set_view()和 set_projection() 三</f64></f64></f64></f64></f64>	器。 y) 设 的变 我们 D矩 三个函
	构成。我们下一步需要做的就是视口变window_height),这些在光栅化器中是如何运作的,这一点十分重要。 其他你们可能关心的函数: compute_barycentric2d(x: f64,已经写好的计算 Barycentric Cooslides 来理解如何利用该函数来实 get_index(&self, x: usize, y: usize) clear(&mut self, buff: Buffer):将每一帧都是重新渲染的,所以会先	E换,将坐标映射到我们的屏幕中(window_width *都已完成,所以不需要担心。但是,你需要去理解这步打 y: f64, v: &[Vector3 <f64>; 3]) -> (f64, f64, f64): 这是rdinates 的函数。你需要阅读 GAMES101_lec9 第5~12 z_interpolated 的插值生成。 e) -> usize: 返回 (x, y) 所对应的像素点的 index。 depth_buf 和 frame_buf 都重新初始化,因为我们的分清空缓存。对于 LAB1,每次旋转都会 flush buffer 再重应当是每帧渲染相同的结果(LAB2的要求3中所要记录的</f64>	操作 是我们 2页 光栅器 重新這
	choose_shader_texture 已经为你提以及 get_projection_matrix 后,不 kr@Krs-MacBook-Pro John-GAMES1 如果你正确实现了前面所述内容,你应你可以替换 file_path 为你指定的图方法,但需要在 Report 中注明。 在 LAB3 结束后,替换 normal 为 t对应的图像。 如果你想要实现Optional部分的 .obj 一般的 .obj 文件: v 0.437500 0.164062 0.765625 v -0.437500 0.164062 0.765625 v 0.500000 0.093750 0.687500 v -0.500000 0.093750 0.687500 vn -0.4988 0.5300 0.6858 vn 0.6667 -0.3333 0.6667 vn -0.6667 -0.3333 0.6667 vn -0.6667 -0.3333 0.6667 vt 0.676379 0.233241 vt 0.360308 0.235899 vt 0.715342 0.265392 f 1/1/47 47/47/47 49/49/47 65/f 50/50/48 48/48/48 2/2/48 66/	提供了匹配 shader 的接口。在实现了 rasterize_trian 你可以在命令行使用默认的 normal shader 进行渲染。 101-rs-LAB3 % cargo run 'file_path' normal rigion rigio	多得到
	OBJ(Object)文件是一种常用的三维息和其他属性。下面是一般OBJ文件的1. 顶点(Vertices):OBJ文件以顶是构成三角形、多边形和其他几何2. 纹理坐标(Texture Coordinates)标(s、t),通常在二维纹理图像3. 法线(Normals):法线是垂直常型、过索引引用顶点、纹理坐标和运染效果非常型、过索引引用顶点、纹理坐标和方线。Materials):OBJ文件中可以质属性,例如颜色、纹理贴图、透OBJ文件使用关键字来标识不同的部分。v:定义顶点坐标。vt:定义过理坐标。vt:定义过理坐标。vn:定义法线。f:定义面,指定顶点、纹理坐标。mtllib:指定材质库文件的路径。usemtl:指定使用的材质名称。这些关键字和其后的数值信息组成了O	点列表开始,每个顶点由其三维坐标(x、y、z)表示。 [图元的基本点。 : 纹理坐标定义了三维模型上的每个顶点对应的纹理映上使用。纹理坐标可以用于在模型表面上贴上纹理图像。三维模型表面的矢量,用于定义每个顶点或面片的方向。要。 的几何拓扑结构,可以是三角形、四边形或多边形。每个点。 从包含材质库文件(MTL文件)的引用,其中定义了模型的度等。 和属性。常见的关键字包括:	顶点 中射坐 。 法 个面通 e的材
	信息,以及其他属性。 当你尝试用 model/spot 文件夹下的其 // for spot_quadrangulated.obj thread 'main' panicked at 'ind 这是因为这些 obj 文件并没有包含 v 含了所有应有的参数(确切地说,符合 在你使用其他Apps生成 obj 文件时,数),如果有必要,你可能需要自己重新 Eg: 简易的 obj 导入例子,用 Phong Code 文件 編輯 选择 重看 转到 运行 线绳 窗口 帮助 Blender ② 文件 编辑 选择 重看 转到 运行 线绳 窗口 帮助 ② ② 《 》 《 》 》 》 图 图 图 图 图 图 图 图 图 图 图 图 图 图	Idex out of bounds: the len is 0 but the index in 参数,而LAB所采用的 spot_triangulated_good.ob utils/load_triangles 函数的要求) 需要注意生成的文件需要三角化(即 f 项后面只跟三个新修改 load_triangles 函数来实现三角形的导入。 Shader 渲染一个猴头	is 0', j 包 参
	# *** *** *** *** *** *** *** *** *** *	PROBLEMS 1 TERMINAL > TERMINAL	× ○ ×
	Code 文件 编辑 选择 重看 转到 运行 终端 窗口 帮助	Output.png — LAB3-TA Output.png — Wasterizer.rs main.rs monkey.obj monkey.mtl Output.png monkey.mtl Output.png monkey.mtl Output.png monkey.mtl Output.png monkey.obj monkey.	
	Part1: You need to finish LAB1/LAB2 Part2: You need to finish LAB3 before You should write a report for these to implemented the LAB, what difficultion and to make sure the codes you have to let students who haven doing and what you've done after reaching the later to make sure your TA knows REFERENCE	before 2023.7.9 23:59. The 2023.7.18 23:59. The report should include: How ies you encountered, and your own innovation (if any ou submit can run successfully. (Submit to Github) It done the project to be able to understand what are ading your report. The into account when calculating the bonus. You require you to do. If you do so, please indicate it is.	you can in the
	UCSB.	raphics And Mixed Environemnt Seminar, Lingqi Ya	