

类异形工厂游戏

November 9, 2023

Abstract

《异形工厂》是一款关于建立工厂，自动创建以及关于形状组合的游戏。玩家在游戏中需要交付符合要求的越来越复杂的形状，在游戏中不断进步，并解锁升级以加快你的工厂生产速度。游戏很轻松，只需建造工厂，布好设施，无需操作即能自动创造出各种各样的几何图形。让我们也来写一个类异形工厂游戏吧！

1 开始之前

WARNING: All of the code for this project must be your own. You may not copy source code from other students or other sources that you find on the web. Plagiarism will not be tolerated.

- 抄袭是被严厉禁止的。游戏必须是自己从第一行代码开始写成的,符合面向对象和泛型程序设计范式的 C++ 程序。推荐使用第三方的 C++ 图形库来完成图形的绘制,键盘鼠标事件的捕获和处理等等内容,这样可以集中精力设计和实现游戏的架构。
- 共分四个阶段，一二阶段不要求有图形界面，第三阶段才要求完成图形界面。第四阶段则是完成前面阶段的拓展功能，或者你当然也可以自己提出并实现新的拓展功能，我们会根据难度进行评分。每阶段依然是两周，我们会在第三阶段验收前面所有基础功能，在第四阶段验收拓展功能。
- GUI 可以用 Qt 之类第三方库实现的图形窗口，也可以是用 ncurses 之类库实现的终端窗口，简而言之就是任意发挥，界面是否漂亮/酷炫不算在基础功能中。但这里我们需要强调，你实现的项目应该有 C/C++ 代码文件，使用完全不涉及 C/C++ 的引擎并不符合本课程要求。
- Question 内容是我们对你们可能问题的预演和回答，并有对本阶段内容的总结。Note 内容是对我们认为可以自由发挥部分的进一步说明，仅供参考。

Question 1.1: 分数构成是什么样的？

各阶段基础功能：70%（阶段一、二中的功能方面的实现占62%，阶段三中的GUI的实现仅占8%，所以各位可以自行斟酌工作量分配。）

各阶段选做内容：20%，每个选做内容的难度不同，会根据实际情况给相应的分数。

代码风格：10%。

2 技术指标:阶段一

我们参考已有的一些异形工厂游戏，制订了这份技术指标。本节介绍阶段一的要求。

2.1 地图

游戏的地图在逻辑上是一个 $m \times n$ 的矩形，存在边界。每个 1×1 的矩形是一个格子，每个格子是一个地块，每个地块下可能有待开采物，每个地块上可以放置各种设备（例如传送带、开采器等）。

你需要在地图中确定一个交付中心，和一些可以开采的地块（地块下有开采物）。要求交付中心是一个矩阵，大小至少是 2×2 ，矩阵的边缘可以连接传送带。

在地图生成时，交付中心和待开采的地块即已确定，无需在游戏过程中放置。

2.2 开采物

开采物是存在于固定地块下的待开采的产物。游戏目标是利用开采器开采地块来得到相应的产物，并通过在地块上放置传送带等设备将产物运送到交付中心。运送特定数量的对应产物到交付中心即可。

开采物有一个价值属性。不同种类的开采物有不同的价值，成功运送一个开采物到交付中心可以获得相应价值的金币。

要求：

1. 至少有两种开采物。开采物需要存在于固定地块下，并在地图上显示出来。
2. 要求一种开采物可以被2.3节所描述的切割机设备切割，另外一种不可以被切割（切割机停止工作）。

2.3 设备

游戏中，我们需要放置设备来完成游戏目标。需要实现的设备有开采器、传送带、切割机、垃圾桶。每个设备都有一个基础属性：速率，也就是每秒可以处理多少个物品。如果将设备放在同一个地块，那么后放置的设备会覆盖先放置的。可以自行设置用什么按键快捷选择设备，但是需要有一个按键，它的功能是调整设备朝向。

开采器：放置在地块上进行开采，每个开采器占一个地块，开采器有一个开采得到的物品的输出方向。

传送带：运送物品，每个传送带占一个地块，传送带有传送方向。

切割机：将开采的物品纵向切开并分别输出。如果只需要其中一半的图形，那么需要使用垃圾桶清除另一半图形，否则切割机停止工作；每个切割机占两个地块，如图所示。

垃圾桶：可以从所有四个方向上输入物品并永久清除它们。

Extension 2.1: 旋转机

将图形顺时针旋转90度。

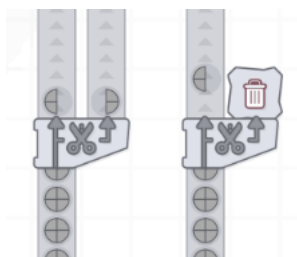


Figure 1: 切割机示例

2.4 Q&A

Question 2.1: 我必须在阶段一实现一个GUI吗?

并非如此。我们只要求项目在最终有一个可用的 GUI，并不强求各位同学在阶段一就实现它的部分或者全部。

不过为了避免不必要的重构，思考如何设计程序的核心逻辑使其能在各种 UI(包括 Graphical User Interface 和 Command Line Interface)上展示是有必要的。

当然，我们认为一边实现核心逻辑，一边实现对应GUI功能是更稳妥的办法。

Question 2.2: 如果我在该阶段没有实现GUI的话，那么我该如何显示呢?

直接在控制台显示日志就可以。

Question 2.3: 我在阶段一必须要实现出什么?

1. 一张可视化地图，显示出交付中心的位置。
 2. 至少两种开采物，并且一种可以被切割机切割、另一种不可以。
 3. 四种设备可以成功放置并实现对应功能。
 4. 工厂可以正常运转，交付中心实时显示当前交付数量。
- 显示可以用日志的格式显示也可以直接使用图形界面。

Question 2.4: 我可以使用 EasyX 、 Unreal Engine 、 NoesisGUI 之类的库吗?

可以,只要它们为程序的开发者提供的 API 是 C++ 的即可。过于复杂和高级的库可能并不容易学习掌握,打包和发布程序或许也会相对困难; 过于简单的库则缺乏一些必要的基础设施,会不必要地加大任务量。

一般来说,我们会推荐初学者使用一些成熟的、有良好文档的、容易上手的库,这样的库不必是 Qt ,或许某些(不为我们所熟悉的)专为游戏开发设计的库是最适合这个项目的。如果有同学对此有心得,欢迎联系我们。

3 技术指标:阶段二

我们已经完成了一个简单的异形工厂游戏! 这听上去棒极了! 所以，让我们为这个游戏添加更多的内容吧！游戏应当基于阶段一中已经完成的部分，只在确实有必要的情况下重构已有的代

码。

3.1 任务设定和局部强化

假设要求有的两种开采物分别为物品1和物品2，其中物品1是设定的可被切割的物品。设定三个任务：成功交付20个物品1、成功交付30个物品2、成功交付50个切割后的物品1。玩家完成一个任务后可以对设备进行升级（开采器的速率、传送带的速率、切割机的速率）。

要求每完成一个任务的时候玩家可以自主选择一个装备进行升级。

注意，这是局部的强化加成，会随着一局游戏的结束而消失。

Extension 3.1: 基于扩展功能的设备（旋转机）设置新的任务

任务四：成功交付30个经过一次旋转的切割后的物品1。

3.2 全局强化系统

通过消耗关卡中收集的金币，玩家可以在游戏主界面的商店中全局强化交付中心的大小、开采物所在地块的数量、地图的大小、每种开采物成功交付后得到的金币数量等。

要求实现以上全局强化内容，你也可以自定义更多的全局强化内容。

注意，这是永久的强化加成，不会随着一局游戏的结束而消失。

3.3 存档的导入与导出

在阶段一中，我们可能每次打开游戏都要重新开始。现在，我们要能让游戏支持存档的导入和导出。你只需为自己设计一套格式用来表达存档，包括全局强化、关卡游戏状态等，演示时修改、加载存档即可。

Note 3.1

被导出和等待导入的存档应当是具有特定语法的文本文件 (可以理解成一种配置文件)。这样，玩家可以:

1. 首先导出当前的存档;

2. 然后可以参考其格式，对其进行修改后成为新的存档;

3. 最后将新的存档导入到游戏中并正常游玩。

各位同学可以自由设计自己的存档的语法。在这个语法下，一个合法的文本文件就唯一对应于一张存档。

3.4 Q&A

Question 3.1: GUI问题?

同阶段一。

Question 3.2: 我在阶段二必须要实现出什么？

1. 任务设定和局部强化系统。
2. 全局强化系统。
3. 存档的导入导出。

显示可以用日志的格式显示也可以直接使用图形界面。

4 技术指标：阶段三

我们已经完成了一个比较复杂的异形工厂游戏，如果你前面没有实现GUI，那么现在是时候加入GUI了。此外，在本阶段，还有一些 (容易使用 GUI 实现的) 实用功能需要完成。

在阶段二中,我们实现了游戏的核心逻辑。但是，如果整个工厂的运作没有一个可视化的表现，那么整个游玩的体验就会被大大损害。

因此，在阶段三中，我们需要在游戏的界面中展示整个工厂的运转效果，具体来讲：

- 开采的物品随着传送带传送的动态效果
- 开采的物品由于设备停止工作或其他原因而在传送带上停滞的表现
- 开采的物品被切割机切割后的变化
- 每个放置的设备的朝向的可视化表现

4.1 Q&A

Question 4.1: 我必须在阶段三实现一个GUI吗？

是的。但如果很喜欢终端窗口的话，也可以用 `ncurses` 之类的库。如果不使用相关的库的话，可能需要自己重新造轮子来完成规定动作。

如果某些动作在已有的程序架构下难以完成，强烈建议换一种展示思路或者实现思路，也可以不实现，但不实现可能会影响项目的分数。

Question 4.2: 我在阶段三必须要实现出什么？

1. 图形界面显示整个工厂的运转效果。

5 技术指标：阶段四

我们已经完成了一个“有趣”的异形工厂游戏啦！为了让它更加的有趣，此阶段是扩展阶段，你可以单纯按照前面的Extension内容来补充，也可以再加上任何你觉得很酷的想法。

比如，你可以为游戏添加适当的音效。从功能上讲，可以设计酷炫的动画来提升用户体验。可以自定义并添加新的设备。任何你觉得有趣的想法都可以加上来！

6 如何提问

6.1 提出问题的错误方式

(不相干的问题)电脑出现了故障应该如何处理。

请联系有能力处理这些故障的人,而不是你的程序设计课的助教。

当然,如果是和编程强相关的、软件方面的故障,而你自己毫无头绪,你当然可以问一问。

(可以在搜索引擎里非常轻松地找到答案的问题) 编译器抱怨发现了一个 `error` 并停止编译

编译器讲的是人话(虽然是英语),真的。我们不想成为复读机,尤其是当问题的答案就在问题的描述里的时候。

当然,如果你相信问题不出在你的代码上,互联网上也没有相关的讨论的时候,欢迎和我们一起探讨。

我的程序又bug了,我怎么办?

原则上,不应该让别人帮你debug,如果实在是无法解决(搜索了全网都没办法解决,折腾了好久),可以寻求TA帮助。不应该一遇到error就找TA寻求帮助。在寻求帮助的时候,最好需要给出你的运行环境,你的错误是什么,你尝试了哪些解决方法。

7 参考资料

7.1 图形库推荐

如果对 C++ 的图形库并不熟悉,这里我们从简单到复杂给出三个库供参考:

FLTK 是一个相当轻量级的 C++ 图形库。

wxWidgets 是一个相对轻量级的 C++ 图形库。

Qt 是非常流行的 C++ 图形应用开发框架。

其中的每一个都有足够多的文档和示例程序,互联网上也有足够多的讨论内容,可供学习使用。

7.2 图形库的安装

对于 FLTK, Windows 用户可能需要安装 MSYS2才能从中得到预编译的库;大部分的 Linux 发行版通过自己的包管理系统提供打包好的库和头文件; macOS 用户可以使用 Homebrew 作为自己的包管理器。当然,总是可以直接下载它的源代码然后自行编译- 这其实不是一件困难的事情,但是想要完全理解整个过程,需要综合运用各位同学在计算机系的专业课(主要是计算机系统基础)上学到的知识。对 wxWidgets 和 Qt 而言,在主流的操作系统下都有打包好的安装程序。

7.3 其他有用的工具

Address Sanitizer 是一个开箱即用的内存安全错误检查工具,它被集成在包括 MSVC 在内的所有主流编译器中。它对于找出 C++ 程序中的内存错误很有帮助。

clang-format 是一个代码排版工具。可以使用它把 C++ 代码排版成自己喜欢的样式,很多主流编译器比如CLion和VS都支持。

clang-tidy是一个代码风格检查工具。它可以就代码的不良风格给出警告。

-Wall 是 gcc/clang 编译器的编译选项。开启它将得到更多来自编译器的警告,这有助于在编写代码的过程中尽早发现问题。类似的选项还有很多,可以参考[这里](#)或者查询编译器的在线手册(manual)。

7.4 Qt教程

<https://www.bilibili.com/video/BV1g4411H78N>

https://blog.csdn.net/qq_46239972/article/details/106862160

https://blog.csdn.net/m0_47055280/article/details/106918193

7.5 关于开源项目

我们不禁止（但是也不推荐）在开源项目基础之上完成该项目，但是在验收的时候需要做到两点：

1. 主动告知使用的开源项目情况；
2. 强调你所做的修改，尤其是和本课程相关的修改。

举一个极端例子，假如一个开源项目已经达到我们的需求，但是如果你重构了它的代码，以更符合面向对象思想实现，那么不会认为是抄袭。

但是如果直接拿这个项目验收而不修改（或者修改很少），那么将会判定为抄袭。

注意：图像素材的替换不会被认为和本课程相关。