芯语教育课程包 · 零基础专项 · 08

零基础通关课程升级方案
16 课 · 从会打开到做出作品

先诊断 12 课断点，再给出逐课讲解脚本、演示步骤与通关检查点

对象	课时	目标	版本
零基础学生 未接触过编程	16 次 × 90 分钟 四阶段递进	课课有检查点 阶段有作品	v1.1 · 2026-10

怎么用这份方案：这是面向零基础的独立 16 课方案，和 02《教师课程包·12 课时》并用。已有编程基础或高校通识班继续用 12 课；零基础班、社团启蒙班、中小学课后服务班按本方案执行。每课都配“讲解脚本—演示步骤—学生任务—通关检查点—卡点救场”，教师可以照着讲、照着做。

一、先把问题说透：12 课对零基础卡在哪里

我们把 02 的 12 课逐课对照“零基础学生第一次接触电脑编程”的真实起点，发现 6 个断点。升级方案就是逐条补上这 6 个断点。

序号	断点	具体表现	升级做法
1	没有“第一次开机”	第 1 课就要求区分 A/B 轨、找四类节点，零基础学生连客户端首页都没认全。	新增第 1—2 课：认识界面、运行、停止、找回窗口；只做“看见结果”这一件事。
2	概念没有具体经验	第 3 课直接讲“变量回写”，学生还没亲手让一个数字变过三次。	第 3 课先“改数字看变化”，第 5 课才正式命名“变量”，先体验后定义。
3	证据要求过高	第 2 课就要求读轨迹、第 4 课要求写真值表，抽象符号一步到位。	轨迹改成“三色贴纸法”，真值表改成“两条路线图”，先画后写。
4	A 轨到 B 轨断崖	第 7 课直接读 3D 贪吃蛇完整源码，缺少单文件、2D 画布的过渡。	第 10—12 课做缓冲：函数 → 2D 画布 → 单文件 9S，再进第 12 课贪吃蛇 3D。
5	出错没有救场	常见错误只“列出”，没告诉学生卡住 90 秒后具体按哪里。	新增“90 秒急救卡”和 12 条错误急救表，贴在课桌角。
6	没有通关门槛	12 课没有一课一检查点，学没学会靠感觉，掉队后无法补。	每课 3 条可观察检查点 + 阶段通关门槛 + 补课包，未过不进下一阶段。

小结：问题不在内容多少，而在起点没铺、台阶太高、出错没人管、学会没标准。升级的重点是补台阶、补脚本、补检查点。

二、16 课总课表（四阶段）

阶段	课次	主题	核心概念	学生产出	通关检查点
一 会打开会运行	1	第一次开机：认识芯语创作客户端	首页、示例库、运行、停止、窗口找回	运行截图 1 张 + 界面 5 个区域标注	能独立打开并运行一个示例
	2	看懂顺序：一句话一句话执行	顺序、输出、等待、舞台	把 3 句输出改成自己的话	能说出“先做什么、后做什么”
	3	会改一处：数字、文字、颜色	参数、只改一处、改后必运行	改造前后对比截图	能只改一处并说清变化
	4	单步与轨迹：程序怎么走	单步、轨迹、执行位置	三色贴纸轨迹图	能预测下一步并验证
二 会读会改	5	变量：程序记住的东西	变量、赋值、输入、回写	三次输入记录表	能指出变量在哪、值是多少
	6	条件：如果 / 否则	比较、布尔、且 / 或、分支	两条路线图	能判断条件真假并走对分支
	7	循环：重复而不失控	当循环、从…到…、次数上限	循环次数实验表	能改上限并解释为什么停
	8	状态：分数、生命、多个对象	累积、列表、增删查	分数预测表	能预测 3 轮后的分数
三 会写会调	9	事件与按键：让作品动起来	按键函数、定时器、帧刷新	按键控制小方块	能解释“按一下发生了什么”
	10	函数：把重复包成一块	函数定义、调用、参数、返回	自写 2 个函数	能调用自己写的函数
	11	2D 画布：自己画界面	画布、画文本、画矩形、颜色	2D 小画板	能画出文字和方块
	12	原生 9S 阅读：贪吃蛇 3D	源码结构、3D 场景、对象列表、音效	源码行对应表 + 4D 截图	能定位 5 个关键 API
四 会做会讲	13	原生 9S 改造：俄罗斯方块 3D / 打飞机 3D	顶部配置区、一次只改一处、VM 验证	改造前后差异记录	能改一处并重新跑通
	14	游戏闭环：碰撞、计分、生命、音效	碰撞判定、计分、生命、音效	打飞机 3D 规则解剖图	能讲清一局游戏的四要素
	15	AI 协作与验收：让 AI 帮你，不让 AI 骗你	需求描述、模板路由、来源验收	提示词 + 验收单	能判断原生 / 预览 / 原型
	16	综合项目：做出并讲清自己的作品	需求、改造、测试、表达	可运行作品 + 3 分钟答辩	作品能运行、能解释、能复现

三、每课 90 分钟标准节奏

环节	时间	教师做什么	学生做什么
开场唤醒	5′	用上一课作品开场，问一个只需一句话回答的问题。	举手回答，打开客户端。
讲解脚本	12′	照本课“讲解脚本”念 3—5 句，不超过 3 个新词。	在任务单上圈出新词。
教师演示	10′	按“演示步骤”做一遍，故意留一个错误并当场修好。	看、记、跟读操作口令。
跟着做	15′	喊口令，全班同步做一遍保底任务。	完成保底任务，举手示意。
自己做	28′	巡视，只对卡住超过 90 秒的学生出手。	标准任务 + 自选挑战任务。
展示与检查	15′	抽 2 组展示，按检查点逐条打勾。	互评、补做、贴通关贴纸。
出口条	5′	发一张 3 行小条，收齐才算下课。	写“我改了 / 我看到的 / 我还想问”。

> **三遍法：**每个新操作至少走三遍：教师做一遍（看），全班跟一遍（仿），学生自己再做一遍（独立）。三遍没走完，先别急着讲下一个概念。

四、16 课逐课教学脚本

01 第一次开机：认识芯语创作客户端

阶段一 · 会打开

本课只做　打开、运行、停止、找回窗口

新词　示例库、运行作品、作品舞台

讲解脚本（照念）

“今天我们不做游戏，只学会三件事：打开它、让它跑、让它停。中间这一块叫作品舞台，所有结果都在这里出现。左边是示例库，点一个名字就能载入。记住两个按钮：运行作品、停止。”

演示步骤

1. 双击桌面“芯语创作客户端”，指着首页说“这是我们的工作台”。
2. 点“示例库”，载入《星际寻宝》，点“运行作品”。
3. 指着舞台上的输出逐句读，再点“停止”。
4. 故意把窗口最小化，演示从任务栏找回。
5. 再运行一次，让学生数“一共出现了几句话”。

学生任务

保底：独立完成一次“载入—运行—停止”，截 1 张图。**标准：**在截图上标出首页、示例库、源码区、舞台、运行按钮 5 个区域。**挑战：**换一个示例再跑一遍，说出两个示例的相同点。

通关检查点

1. 不看别人，能自己打开客户端并载入示例。
2. 能指出作品舞台，并说出“结果在这里出现”。
3. 能按“停止”，再从任务栏找回窗口。

卡点救场

图标找不到：从开始菜单搜“芯语创作客户端”。点了没反应：看任务栏是否已有窗口；仍无则重启客户端。舞台空白：先点“运行作品”，再看是否停在“源码”标签。

02 看懂顺序：一句话一句话执行

阶段一 · 会打开

本课只做 顺序、输出、等待

新词 顺序执行、输出、等待

讲解脚本

“程序像排队打饭，一句做完才做下一句，这就叫顺序执行。输出就是让舞台说话。等待就是让程序停一下再继续。今天不改逻辑，只改说话的内容。”

演示步骤

1. 打开《星际寻宝》源码，从上到下逐行指着读。
2. 把第一句输出改成“大家好，我是小芯”，运行。
3. 在两句之间加一个等待，运行并让学生感受停顿。
4. 把顺序颠倒，运行，问“结果顺序变了吗”。
5. 恢复原样，强调“改完必须再运行”。

学生任务

保底：改 1 句输出为自己想说的话并运行成功。**标准**：写 3 句有先后关系的输出，说明为什么不能颠倒。**挑战**：加一个等待，让最后一句延迟出现。

通关检查点

1. 能用自己的话说出“顺序执行”。
2. 改的 1 句输出能成功显示。
3. 能按“先……再……最后……”复述自己程序。

卡点救场

改了没变化：没点运行或没保存。文字全是拼音：输入法没切中文。报错红字：先检查引号是否成对。

03 会改一处：数字、文字、颜色

阶段一 · 会打开

本课只做 参数、只改一处、改后必运行

新词 参数、对比

讲解脚本

“程序里能改的数字、文字、颜色，都叫参数。今天立一条规矩：一次只改一处。改完必须运行，把改前和改后放在一起看，这叫对比。”

演示步骤

1. 载入《俄罗斯方块 3D》，找到顶部配置区的颜色和下落帧。
2. 只改一个颜色，保存，原生运行，截图。
3. 恢复，再只改下落帧，运行，说明手感变化。
4. 同时改两处，运行，问“现在知道是哪一处起作用吗”。
5. 回到“只改一处”，把方法写进任务单。

学生任务

保底：只改一个颜色并运行成功，交改造前后截图。**标准**：再改一个数字参数，用一句话说明变化。**挑战**：给同伴设计“猜我改了哪一处”的小游戏。

通关检查点

1. 能指出顶部配置区在哪里。
2. 只改一处并能说清前后差异。
3. 知道“同时改两处无法判断因果”。

卡点救场

3D 窗口没出现：看任务栏，或点“停止”后重开。改完报错：先恢复上一版，再只改一个字重试。找不到配置区：拉到源码最上方，找“可改配置”注释块。

04 单步与轨迹：程序怎么走

阶段一 · 会打开

本课只做 单步、轨迹、执行位置

新词 单步、轨迹、当前执行

讲解脚本

“运行是一口气跑完，单步是一步一停。每停一次，轨迹就记一笔：走到哪、变量是多少。会看轨迹，你就有了证据，不再靠猜。”

演示步骤

1. 载入《猜数字》，点“单步”，指给学生看高亮位置。

2. 连点 3 次“下一步”，每次让学生先猜变量值。
3. 打开“轨迹”，找到刚才 3 步的记录。
4. 改用连续运行，对比“一次跑完看不到中间”。
5. 发三色贴纸：绿=执行、黄=变量、红=没懂。

学生任务

保底：单步 3 次并贴出 3 张轨迹贴纸。**标准：**先预测再验证，记录 1 次预测错误。**挑战：**用轨迹解释“为什么第 4 步进了如果分支”。

通关检查点

1. 能独立进入单步模式。
2. 能在轨迹里找到“当前执行”那一行。
3. 能说出一次“我猜错了，实际是……”。

卡点救场

下一步是灰的：程序在等输入或已结束，先看舞台提示。轨迹空白：先点“轨迹”标签再单步。单步太快：每步之间让学生把手举起来再放下。

05 变量：程序记住的东西

阶段二 · 会读

本课只做 变量、赋值、输入、回写

新词 变量、赋值、输入

讲解脚本

“变量就是一个有名字的盒子，里面装一个值。赋值就是往盒子里放东西。输入就是让用户往盒子里放。盒子只有一个，所以新值会盖掉旧值。”

演示步骤

1. 在黑板画 3 个盒子：分数、猜测次数、玩家猜测。
2. 运行《猜数字》，输入 9，指给学生看“玩家猜测”变成 9。
3. 再输入 2，强调旧值被盖掉。
4. 打开轨迹，找到三次“猜测次数”的变化。
5. 把初始值改成别的数，再跑，问“哪个盒子先变”。

学生任务

保底：完成 9、2、7 三次输入并填记录表。**标准：**指出三个变量各自在第几步变化。**挑战：**设计 3 个输入，让“猜测次数”刚好到 3。

通关检查点

1. 能指出一个变量名和它当前的值。

2 能说出“输入会写进哪个变量”。

3 能解释“为什么新值会盖掉旧值”。

卡点救场

输入字母没反应：数字框只收数字。变量看不到：答案变量被隐藏，看轨迹或调试面板。记录表不会填：只填“输入值、变量值、输出”三列。

06 条件：如果 / 否则

阶段二 · 会读

本课只做 比较、布尔、分支

新词 条件、真、假、分支

讲解脚本

“条件就是一句能回答真或假的话，比如‘猜测大于答案’。真，走如果这一边；假，走否则这一边。两条路只能走一条，这就是分支。”

演示步骤

1. 黑板写两条路：左“大了”，右“小了”，中间写条件。
2. 输入 9，让学生判断条件真假，再看程序走哪边。
3. 输入 2，走另一边；输入答案，走“猜中”。
4. 把“大于”改成“小于”，运行，观察提示反了。
5. 恢复，强调“先写中文，再翻成条件”。

学生任务

保底：画出两条路线图，标出 3 次输入各走哪边。**标准：**完成 3 行真假记录。**挑战：**加一条“输入 0 提示超范围”的分支。

通关检查点

1 能说出一个条件是真还是假。

2 能画出如果 / 否则两条路。

3 知道“=”是赋值，“==”才是比较。

卡点救场

提示反了：比较符号写反。条件永远为真：检查是否用了“=”。嵌套看不懂：先拆成两个独立的如果。

07 循环：重复而不失控

阶段二 · 会读

本课只做 当循环、次数上限、退出

新词 循环、继续条件、退出

讲解脚本

“循环就是反复做同一件事。但必须回答两个问题：什么时候继续？什么时候停？停不下来叫死循环，运行环境会在 200 次时强制刹车。”

演示步骤

1. 让 3 个学生上台，模拟“猜错就再来一次”。
2. 打开循环条件，指出两个退出条件：猜中、次数用完。
3. 把上限 5 改成 3，连续输错，看第 4 次是否还被允许。
4. 故意删掉“次数 +1”，运行，观察安全保护。
5. 恢复，强调“循环体里必须让状态变化”。

学生任务

保底：把上限改成 3 并验证。**标准：**填循环次数实验表，写出退出原因。**挑战：**设计“3 次内猜中给奖励”的规则。

通关检查点

1. 能指出循环的继续条件和退出条件。
2. 改上限后能验证循环提前结束。
3. 能说出死循环为什么会发生。

卡点救场

改了上限没生效：源码和积木输入框没同步。循环不停：检查循环体里有没有更新次数。程序卡住：点停止，先从“次数 +1”查起。

08 状态：分数、生命、多个对象

阶段二 · 会读

本课只做 累积、列表、增删查

新词 状态、列表、长度

讲解脚本

“分数不会每轮清零，它跨轮保留，这叫状态。游戏里有多个敌机、多颗子弹，就用列表来装：追加是放进去，移除是拿出来，长度是数一数。”

演示步骤

1. 运行《星际寻宝》，问“分数是重新开始还是一直加”。

2. 把轮次改 2、分值改 5，先让学生算，再运行。
3. 再改 4 轮、2 分，验证预测。
4. 打开打飞机 3D 源码，找到“敌机ID”列表和“列表长度”。
5. 用黑板画一个抽屉，演示追加、移除。

学生任务

保底：完成 2 轮 × 5 分并预测正确。**标准：**填“轮次 × 分值 = 总分”表 3 行。**挑战：**在打飞机源码里找出 3 个列表并说明各装什么。

通关检查点

1. 能预测循环结束后的总分。
2. 能指出一个列表和它的长度。
3. 能区分“状态”和“临时变量”。

卡点救场

预测总错：先只改轮次，再只改分值。列表看不懂：先只记“装东西的抽屉”。变量名混：把两个名字写在便利贴上贴屏幕边。

09 事件与按键：让作品动起来

阶段三 · 会写

本课只做 按键函数、定时器、帧刷新

新词 事件、定时器、帧

讲解脚本

“按键不是程序自己动的，是你按了它才动，这叫事件。定时器让程序每隔几帧自动做一次，这叫刷新。一个作品 = 按键管你，定时器管世界。”

演示步骤

1. 打开贪吃蛇 3D，按方向键，指着蛇头说“这一下是谁通知了程序”。
2. 找到“设置按键函数”，读出按键分支。
3. 找到“设置定时器”，说明每隔几帧移动一格。
4. 把移动间隔改大，运行，观察变慢。
5. 恢复，画出“按键→函数→画面”的箭头图。

学生任务

保底：画出“按键→动作→画面”三格箭头图。**标准：**改一次移动间隔并说明快慢。**挑战：**在打飞机 3D 里找出“自动开火”对应的定时逻辑。

通关检查点

1. 能说出“按键触发函数”。

2 能指出定时器在源码哪一段。

3 知道“帧”是画面刷新的单位。

卡点救场

按键没反应：3D 窗口没获得焦点，先点一下窗口。蛇不动：定时器没启动或游戏结束。改了间隔没变：保存后再运行。

10 函数：把重复包成一块

阶段三 · 会写

本课只做 函数定义、调用、参数

新词 函数、调用、参数

讲解脚本

“同一段话写三遍，不如包成一个函数，起个名字，用的时候喊名字。名字后面括号里的东西叫参数，是每次可以换的值。”

演示步骤

1. 黑板写“移动战机”在三处重复出现，圈出来。
2. 指出源码里的“函数 移动战机()”定义。
3. 演示调用一次，画面变化。
4. 加一个参数“距离”，让调用时可以换值。
5. 说明“定义一次，到处调用”。

学生任务

保底：找到并调用一个现成函数，让画面变化。**标准：**写一个只含 2 句输出的函数并调用 2 次。**挑战：**给函数加 1 个参数，用两个不同值调用。

通关检查点

1 能指出“定义”和“调用”的区别。

2 能调用自己写的函数并看到结果。

3 能说出一个参数的作用。

卡点救场

调用没反应：函数名拼错或没定义。参数对不上：个数和顺序要一致。函数里输出不显示：检查是否真的调用了。

11 2D 画布：自己画界面

阶段三 · 会写

本课只做 画布、画文本、画矩形、颜色

新词 画布、坐标、颜色值

讲解脚本

“3D 之前先学 2D。画布就是一块白板，坐标决定画在哪，颜色值决定什么颜色。所有绘制都要先拿到画布，再在画布上画。”

演示步骤

1. 运行一个 2D 示例，指着左上角说“这里是 0,0”。
2. 演示画一行文字，改坐标，看位置移动。
3. 演示画一个矩形，改宽高和颜色。
4. 指出代码里的“九S画布()”作为第一个参数。
5. 故意漏掉画布参数，展示报错，再补上。

学生任务

保底：画出一行文字和一个方块。**标准**：把方块移到屏幕右下角。**挑战**：用 3 种颜色画 3 个方块并排。

通关检查点

1. 能说出坐标原点在哪。
2. 能画出文字和矩形。
3. 知道绘制函数第一个参数是画布。

卡点救场

什么都不显示：检查是否真的调用了绘制、是否在刷新帧里。位置不对：坐标顺序是 X 再 Y。颜色不生效：颜色值要用数值，不是文字。

12 原生 9S 阅读：贪吃蛇 3D

阶段三 · 会写

本课只做 源码结构、3D 场景、对象列表、音效

新词 3D 场景、相机、渲染

讲解脚本

“从今天起我们读完整源码，它由芯语虚拟机直接运行，没有积木视图。读源码只看四件事：场景怎么建、对象怎么加、每帧怎么动、什么时候结束。”

演示步骤

1. 打开贪吃蛇 3D，先只看顶部配置区。
2. 找到“创建3D场景”，圈出相机、光源两行。

3. 找到“创建蛇块”，说明每个蛇块是一个 3D 物体。
4. 找到“设置按键函数”和“设置定时器”。
5. 启动 4D 虚拟机玩一局，再回来对应源码。

学生任务

保底：填 5 行“源码行—功能—窗口表现”对应表。**标准**：改蛇身颜色并重新运行成功。**挑战**：解释“列表”为什么能表示整条蛇。

通关检查点

1. 能定位创建 3D 场景、相机、渲染三处。
2. 能启动 4D 窗口并正常游玩。
3. 能说出源码某一行对应窗口哪个变化。

卡点救场

找不到积木：B 轨本来就没有，这是设计边界。窗口没出现：看任务栏，或点停止后重开。改完跑不起来：先看资源路径和引号。

13 原生 9S 改造：俄罗斯方块 3D / 打飞机 3D

阶段四 · 会做

本课只做 顶部配置区、一次只改一处、VM 验证

新词 配置区、回归验证

讲解脚本

“改造不是重写，是找到一个开关，拨一下，再跑一遍。顶部配置区就是给初学者准备的开关面板。改完能跑通、画面有变化，才算改造成功。”

演示步骤

1. 打开俄罗斯方块 3D 顶部配置区，逐行读颜色、速度、开关。
2. 只改方块颜色，保存，4D 运行，截图。
3. 打开打飞机 3D，读“玩家速度、子弹速度、开火间隔、初始生命”。
4. 只改“初始生命”，运行，观察生命数。
5. 恢复原值，强调“记录改前改后”。

学生任务

保底：完成一次颜色改造并跑通。**标准**：完成一次数字参数改造并写差异说明。**挑战**：做“改两处但分开验证”的对照实验。

通关检查点

1. 能找到顶部配置区并读懂 5 个配置项。

2. 改一处后能重新跑通。
3. 能提交改造前后差异记录。

卡点救场

改坏无法运行：恢复原值，再一次只改一处。改了没效果：确认改的是正在运行的那份源码。窗口重叠：先停止再运行。

14 游戏闭环：碰撞、计分、生命、音效

阶段四 · 会做

本课只做 碰撞判定、计分、生命、音效

新词 碰撞、得分、生命值

讲解脚本

“一个能玩的游戏要有四件事：打中了没有、加不加分、撞了扣不扣命、有没有声音。打飞机 3D 把这四件事写全了，我们拆开看。”

演示步骤

1. 玩一局打飞机 3D，指着得分和生命。
2. 找到“检测碰撞”，读出子弹与敌机的差值判断。
3. 找到“分数 = 分数 + 10”，说明计分。
4. 找到“生命 = 生命 - 1”和结束条件。
5. 找到“播放音效文件”，说明开火、爆炸、结束三种声音。

学生任务

保底：画出“子弹→敌机→加分→声音”闭环图。**标准：**把击中得分从 10 改成 20 并验证。**挑战：**把初始生命改成 2，体验难度并说明。

通关检查点

1. 能说出碰撞判定的三个方向差值。
2. 能改分值并看到分数变化。
3. 能说出音效在哪些时刻播放。

卡点救场

打不中：敌机与玩家距离设置太远，先别改判定范围。没声音：确认运行的是 4D 虚拟机且资源目录存在。分数不涨：检查子弹列表是否真的删除了敌机。

15 AI 协作与验收：让 AI 帮你，不让 AI 骗你

阶段四 · 会做

本课只做 需求描述、模板路由、来源验收

新词 提示词、模板路由、验收

讲解脚本

“AI 不是替你写完整游戏，它先判断你的需求该用哪个模板，再帮你改。你要做的是说清楚需求，然后验收：来源是什么、能不能跑、哪里是 AI 改的。”

演示步骤

1. 对比两条需求：“做个游戏”与“做一个能开火、能计分的 3D 打飞机”。
2. 用后一条生成，看界面显示的来源标签。
3. 指出“原生 9S 源码模板”与“内嵌预览”的区别。
4. 打开验收单，逐项打勾：来源、运行、中文、AI 声明。
5. 把一条不合格的描述改写成可验收描述。

学生任务

保底：完成一次模板路由并填验收单 4 项。**标准**：写一条包含目标、规则、反馈、限制的需求。**挑战**：识别一条“把预览说成原生”的错误说明并纠正。

通关检查点

1. 能写出 4 要素齐全的需求。
2. 能判断结果是原生模板、内嵌预览还是教学原型。
3. 能完成一份 AI 使用与来源声明。

卡点救场

生成结果不对：需求里补上规则和限制。分不清来源：看来源标签和“完整 9S 源码”入口。AI 声明不会写：按“用了什么、改了哪里、怎么验证”三句写。

16 综合项目：做出并讲清自己的作品

阶段四 · 会做

本课只做 需求、改造、测试、表达

新词 可运行、可解释、可复现

讲解脚本

“最后一课不看谁做得大，看谁讲得清。作品要过三关：能运行、能解释一处关键代码、别人拿到能复现。三个人一小组，互相当评委。”

演示步骤

1. 展示一份合格作品文档：目标、玩法、来源、AI 声明、限制。

2. 演示 3 分钟答辩结构：目标—演示—一处代码—一个困难—下一步。
3. 用检查表现场给示例作品打分。
4. 让学生互测：把作品发给同伴，按说明运行。
5. 收齐作品包、答辩记录、反思条。

学生任务

保底：提交一个能运行的作品 + 3 分钟答辩。**标准**：指出一处自己改的代码并解释因果。**挑战**：根据同伴反馈完成一次迭代并复测。

通关检查点

1. 作品能在教师机上运行。
2. 能指出并解释至少一处自己改的代码。
3. 同伴按文档能复现。

卡点救场

作品跑不起来：回到上一版可运行状态，再逐步加回改动。讲不清：只说“我改了哪里、画面哪里变了”。不能复现：补运行说明和资源文件。

五、六条课堂做法（以通关为目标）

课前一分钟自检

每课开始前，学生按 4 项清单自检：客户端能开、示例能载入、舞台有画面、输入法中文。任何一项不通过，先修环境再上课。

90 秒急救卡

卡住先等 90 秒，按“看提示—回到上一版—只改一处—举手”四步自救。教师只对超时学生出手，避免全班依赖。

一课一检查点

每课 3 条可观察检查点，全部打勾才贴“通关贴纸”。未过者进入补课包，不带着漏洞进下一阶段。

三层任务

保底任务人人必做，标准任务大多数完成，挑战任务供学有余力者。评价只看“保底 + 标准”，挑战只加分不扣分。

阶段门槛

阶段一过“能运行会改一处”，阶段二过“能预测能解释”，阶段三过“能写函数能读源码”，阶段四过“作品能跑能讲”。

同伴互测

每阶段末交换作品，按说明运行。跑不起来，先补运行说明和资源文件；同伴互测看的是文档能否让别人复现。

六、课堂错误急救表（贴课桌角）

序号	现象	先做什么	还不行怎么办
1	点运行没反应	看是否在“源码”标签、是否已保存	停止后重新运行
2	舞台空白	向下滚动到作品舞台	换一个示例验证环境
3	改了没变化	确认改的是当前项目并保存	关闭重开项目
4	红字报错	先查引号、括号是否成对	恢复上一版，只改一处
5	输入没反应	确认是数字输入框、按了回车	改用鼠标点确认按钮
6	3D 窗口不见	看任务栏、点“原生运行”	停止后重开；确认 4D 虚拟机
7	3D 里按键无效	先点一下 3D 窗口获得焦点	关闭窗口重新运行
8	没有声音	确认 4D 运行、资源目录存在	换一台机器对比
9	循环停不下来	点停止，查循环体是否更新次数	恢复上一版
10	分数不增加	查加分语句是否在碰撞分支内	用轨迹确认是否进入分支
11	文字变拼音	切换中文输入法	先用英文验证，再补中文
12	作品发给别人跑不了	补运行说明和资源文件	导出完整 `.9s` 或工程包

七、评价与通关标准

维度	通关（合格）	良好	优秀
能运行	能独立打开并运行示例	能改造一处并跑通	能独立完成一个小作品
能解释	能说出先后顺序	能解释一个条件或循环	能用轨迹/源码行举证
能迁移	会改颜色、数字	会改规则参数	能把方法用到新模板
能表达	能交截图	能写改造说明	能 3 分钟答辩并接受追问
能协作	能按说明运行同伴作品	能给出 1 条有效反馈	能根据反馈迭代并复测

通关规则：四阶段各自设一道门槛，阶段内 16 课允许 2 课补考。补考只考“保底任务 + 检查点”，不考挑战任务。这样既守住底线，又不把零基础学生挡在门外。

八、与现有 12 课、8 课压缩的对应

本方案	对应 02 的 12 课	说明
第 1—2 课	新增	补“第一次开机”和“顺序执行”，零基础必需。
第 3—4 课	第 1—2 课	把“改一处”和“单步轨迹”拆开，降低抽象坡度。
第 5—8 课	第 3—6 课	变量、条件、循环、状态，节奏放慢一倍，每课只讲一个概念。
第 9—11 课	新增	补事件、函数、2D 画布，填平 A 轨到 B 轨的断崖。
第 12—14 课	第 7—8、10 课	贪吃蛇 3D 阅读、两款 3D 改造、游戏闭环解剖。
第 15—16 课	第 9、11—12 课	AI 验收、作品表达与答辩，压缩但不省略。

九、教师备课清单（每课 10 分钟搞定）

1. 打开本课对应示例，确认能运行；3D 课确认 4D 虚拟机窗口能弹出。
2. 把这节课的“讲解脚本”读一遍，控制在 3 个新词以内。
3. 准备一处“故意改错”的演示，让学生看到出错和修复。
4. 打印本课任务单和检查点表，每人一份。
5. 预设两个可能卡住的点，把急救表对应条目写在黑板角。
6. 确认补课包位置：上一课保底任务 + 检查点，供掉队学生课后补。

十、给家长和学生的一张说明

1. **不要求先懂代码。**先会运行、会改一处，再慢慢懂原理，这是本课程的设计顺序。
2. **每节课都有检查点。**三个小动作做到了才算过关，不靠“感觉学会了”。
3. **课程结束交一个作品。**第 16 课每位学生提交一个可运行、能讲清的作品；基础弱的学生可以走保底路径，完成度按检查点评价。

> **边界说明：**本方案是教学建议，不表示官方审定，也不承诺统一的通关结果。16 课按每课 90 分钟设计；如学校为 40—45 分钟一节，请把每课拆成两节，按“讲解+跟做”与“自己做+展示”分开，不要压缩练习时间。