

ICode 太空探险挑战赛规则

(一) 参赛对象：1-10年级中小學生

(二) 比赛形式：线上+线下

(三) 组别设置：

主题：ICode 太空探险挑战赛			
语言 \ 学段	小学低龄组 (1-3 年级)	小学高龄组 (4-6 年级)	中学组 (7-10年级)
图形化	✓		
Python		✓	✓
C++			✓

(四) 题量及时间设置：

每个组别各30 题，须在60 分钟内完成。每个组别统一开始，统一结束，一旦比赛时间结束，将不能再运行和提交代码。

(五) 竞赛平台及相关说明

1. 下载地址

直接使用浏览器打开网页地址并登录即可参加比赛，登录地址为：

<https://c.icode.org.cn/gzht>

2. 安装环境要求

适用于win7 以上系统，Chrome 浏览器。

3.编程语言和界面

(1) 编程语言

主要使用图形化、Python 和编码语言。

(2) 界面

	<pre>for i in range(3): Dev.step(4) Dev.turnRight()</pre>	<pre>for(int i=0; i<3; i++) { Spaceship.step(2) Dev.step(4) }</pre>
图形化编码语言	Python 编程语言	C++编程语言

4.任务场景和任务模型

竞赛内容以任务关卡形式呈现，选手需要使用C++代码，控制关卡中的机器人行动，完成规定的任务。下述展示了其中一个关卡和对应的答案：

	
关卡设置	图形化编码答案
<pre>for i in range(4): Dev.step(5) Dev.turnLeft()</pre>	<pre>for(int i=0; i<4; i++) { Dev.step(5) Dev.turnLeft() }</pre>

5.练习与竞赛场景说明

(1) 练习

选手可以登录以下学习平台进行赛事内容的学习和练习。通用学习平台: <https://home.icode.org.cn/>

(2) 竞赛场景

比赛形式为线下比赛。选手报名比赛后，登录比赛平台，输入账号密码即可开始比赛。

(六) 竞赛分组及设置

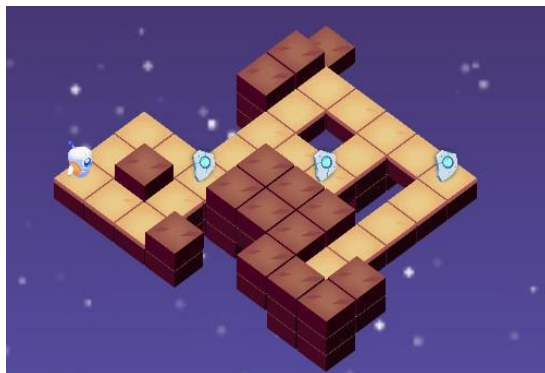
1.竞赛分组

竞赛将分为多个组别：

图形化小低组（1-3 年级），Python小高组（4-6 年级），Python 中学组（7-10 年级），C++中学组（7-10 年级）。各个组别分别进行评奖。

2.竞赛内容设置

竞赛内容以任务关卡形式呈现，选手需要使用积木代码、Python代码、C++，控制关卡中的机器人行动，完成规定的收集任务。



任务关卡示意图

在各组别中，任务关卡所设计的知识点如下表所示：

图形化	Python	C++	备注
ICode基本操作	程序的顺序执行	程序的顺序执行	
顺序执行	简单for循环	简单循环	
重复执行	循环和变量	循环和变量	
嵌套循环	列表的遍历与使	数组和遍历	仅小学高年级、 中学组考察
条件判断	While循环	While循环	
函数	嵌套结构（循环和条件）	循环判断	仅中学组考察
综合应用	函数应用	函数	
	递归	递归	
	综合应用	综合应用	

（七）竞赛任务

竞赛内容以任务关卡形式呈现，选手需要使用图形化代码、Python代码，控制关卡中的机器人行动，完成规定的任务。

（八）评分标准制定原则、评分方法、评分细则

1.竞赛流程

(1) 在比赛官网：www.gzcpss.com及官方公众号“科普赛事”，查看有关该赛事发布的文章，联系相关的工作人员，了解赛事及报名信息；

(2) 根据指导完成报名后，在规定的比赛日期及时间内，使用比赛账号登录比赛平台，参加比赛；

(3) 每一轮比赛结束后，组委会对比赛成绩进行核实，并公布。

2. 关卡评星规则

竞赛平台将会按照任务完成的情况自动进行评分，每个关卡最高可获得3分，评分规则如下：

(1) 完成关卡的任务，得到1分；

(2) 在完成关卡任务的基础上，如果使用的代码行数少于等于指定行数，额外获得1分；

(3) 在完成关卡任务的基础上，如果角色的移动步数少于等于指定步数，额外获得1分；

(4) 比赛期间，参赛选手所有关卡的得分总和，即为该选手最终得分。同时，若选手的最终得分相同，根据选手代码的优化行数和优化步数进行排名。

3. 计分及赛制

在竞赛中，选手排名将会显示在排行榜中。排名规则如下：

(1) 选手排名将首先参考选手获得的星星数量，星星越多，排名越靠前；

(2) 若两位选手星星数相同，则参考选手优化的行数，优化行数多的排名靠前；

(3) 若选手的前两项标准都相同，则参考选手优化的步数，优化步数多的排名靠前；

(4) 若所有标准均相同，则根据选手数据库记录的提交时间排名，优先提交的选手排名靠前。

(5) 每个参赛选手的排名依据各选手成绩的分数高低进行排名，同等分数下选手的优化步数和优化行数多的排名靠前。

(6) 如参赛队员对竞赛过程及结果存在异议，现场裁判不能解决的，由参赛队员提交申诉书，交仲裁委员会处理。

(九) 犯规及取消竞赛资格

1.为了竞争得利而作弊是犯规行为，情节严重者可能会被取消比赛资格；

2.参赛选手须独立完成题目，不允许互相抄袭及作弊，一经发现将严肃处理；

3.参赛选手不可尝试使用违规代码完成任务，不可尝试使用技术手段破解或攻击比赛平台，不可使用不合理的手段修改比赛排名数据，若发现此类情况，将取消选手的成绩，情节严重者将被取消参赛资格；

4.禁止冒名顶替参赛，违反者将直接取消参赛资格；

5.赛事组委会将通过多种技术手段监测比赛中出现的异常情况并判定其是否违规，组委会对于违规行为的判定和处理拥有最终解释权。

6.其他违例细则按照“竞赛总则”执行。

(十) 大赛声明

1.大赛本着公益性、自愿参赛的原则，由选手自主决定是否参与。

2.成功完成比赛报名的参赛选手，请遵照大赛官网官的通知，在规定时间内完成参赛证的下载与打印，并按时参加比赛。

3.如选手在报名参赛期间出现作弊等违规行为，组委会有权取消其考试资格，并在大赛官网通报批评。

4.如因不可抗力因素导致的大赛延期或停办，组委会将不承担任何责任。

5.本次大赛最终解释权归赛事组委会所有。

001Code人工智能编程闯关挑战赛

活 动 规 则

目 录

活动对象	3
主题设置	3
分组情况	3
报名要求	3
报名方式	4
一、活动内容	5
二、活动平台	6
三、任务及规则	6
四、活动方式及计分	7
五、活动设置	8
六、犯规及取消活动资格	9
七、其他说明	9

活动对象

活动对象为各地青少年。各学校、科技馆及其他科普专业机构、各教育机构需做好青少年学员组织工作，协助其集体参加线上线下展示活动。

主题设置

人工智能编程闯关挑战赛

分组情况

小学低年级组 1-3 年级（图形化）

小学高年级组 4-6 年级（图形化/Python/C++）

初中组（Python/C++）

高中组（Python/C++）

报名要求

本活动项仅支持个人报名，报名人数 1 人，以报名系统显示的要求为准。

报名方式

学生登录活动官网→选择“人工智能编程闯关挑战赛”
→点击“报名”按钮，获取报名和参加活动的链接，非通过
该渠道的报名无效，以报名系统验证的兼报规则为准。

一、活动内容

1.1 活动介绍

编程是信息及人工智能领域里最重要的技能，本项目将带领活动学生踏上技术与场景交融的探索之旅。活动学生需通过图形化或 Python 语言、C++语言编写代码，控制角色完成路径规划、资源收集、障碍避让等任务目标。

1.2 活动主题

本次活动以“人工智能编程闯关挑战赛”为概念主题，聚焦人工智能前沿技术与未来社会应用的深度融合，旨在激发青少年对 AI 技术的探索热情与创新能力，鼓励学生用编程解决真实问题，探索人工智能等高新前沿领域。

1.3 活动要求

本项目分为 4 个组别 3 种编程语言：

小学低年级组 1-3 年级（图形化）

小学高年级组 4-6 年级（图形化/Python/C++）

初中组（Python/C++）

高中组（Python/C++）

1.4 环境要求

系统：Windows 7 sp2（MacOS 10.13）及以上版本；

浏览器：Chrome 109.x 及以上版本；

内存：不少于 8GB；

硬盘：可用容量不低于 32GB；

网络：单台次上下行带宽不小于 20Mbps；

二、活动平台

2.1 练习

活动**不限制**练习平台，学生可以根据报名时选择的图形化、Python 或 C++自行选择对应的平台进行练习。

2.2 活动前模拟

活动正式开始前，将公布和开启活动平台，学生应提前登录活动平台进行注册和绑定，并选择对应的编程语言，学生可以选择 1 次编程语言，初审开始后至省级展演结束，均无法修改编程语言。

平台活动前将开放至少 2 次的模拟活动环境，主要用于测试账号，模拟活动的成绩与正式活动无关，每个学生账号应当进行模拟活动，但未参与模拟活动不影响正式活动。

2.3 活动

鼓励有条件的地区，由区域组委会组织到机房参与活动；未组织线下活动的地区，参加统一线上活动。

三、任务及规则

3.1 任务说明

活动任务以关卡形式呈现，学生使用图形化或 Python 语言或 C++语言编写程序，控制角色完成指定任务。

3.2 题量及时间

每个组别正式活动题量 30 题，活动时间限制 60 分钟。

每个组别及语言类型的学生统一时间开始及统一时间结束活动，学生可以提前交卷，或待活动时间结束时，系统自动收卷。

3.3 组别、语言

首次登录系统时，学生根据组别及语言限制要求，自主选择使用图形化或 Python 或 C++语言编程语言，编程语言类型仅可选择 1 次，确认后不可修改。

3.4 知识模块

图形化：编程环境基础、顺序结构、循环结构、分支结构、变量及函数的定义及使用等；

Python：顺序结构、for 循环、变量、list 定义及使用、分支结构、while 循环、嵌套循环、函数、递归等；

C++：数据类型（基本类型、自定义类型）、变量与常量定义及使用、顺序结构、if-else 分支结构、switch 分支结构、for 循环、while 循环、数组定义及使用、函数定义。

四、活动方式及计分

4.1 活动方式

本项目分为初选和终评 2 个层级，学生在每个层级中有 1 次活动机会，按计分规则排名择优晋级。

4.2 计分规则

学生运用图形化或 Python 编程语言或 C++ 语言在限定时间内，通过编写程序代码，引导角色顺利完成路径规划、资源收集、障碍避让等任务。

路径规划：使用代码引导角色通过迷宫；

资源收集：收集指定数量的物品；

障碍避让：避开障碍物顺利到达目标；

五、活动设置

5.1 排名规则

分数由系统自动完成，按以下四个维度进行排名：

(1) 完成数量：完成关卡数量越多，排名越靠前；

(2) 代码行数：完成数量一致的情况下，优化后总行数越少，排名越靠前；

(3) 代码步数：完成数量、代码优化一致的情况下，优化后总步数越少，排名越靠前；

(4) 活动用时：完成数量、代码优化、步数优化一致的情况下，活动用时越短，排名越靠前；

5.2 结果

根据各组别、语言类型排名情况，按活动安排公布活动

结果。

六、犯规及取消活动资格

6.1 舞弊行为

学生存在交头接耳、左顾右盼、打手势、做暗号、夹带、旁窥、抄袭或有意让他人抄袭、传抄答案、交换账号、替考等行为时，经监考人员确认的，取消其活动资格。

6.2 违规代码

学生使用违规代码或技术手段破解平台或绕过规则的，情节严重的，取消其活动资格。

七、其他说明

在确保公平、公正的前提下，经总裁判长、副总裁判长、活动项裁判长合议，可根据现场实际情况，对活动项有关候场时间、备活动时间、轮候场次等细节问题进行调整，但不得影响计分规则。

附录 1: 《学生成绩单》

学生成绩单

学生姓名		报名编号	
学生组别		语言类型	<input type="checkbox"/> 图形化 <input type="checkbox"/> Python <input type="checkbox"/> C++
完成数量 (1)		代码行数 (2)	
代码步数 (3)		活动用时 (4)	
最终排名			

填写人:

填写日期、时间:

学生签名:

青少年安全与应急创新大赛

AI 应急创新编程设计

一、 比赛主题

应急安全是维护社会稳定、保障人民生命财产安全的重要基石。为全面提升青少年安全素养与应急能力，本竞赛构建了“知识普及、技术应用、创新设计”三位一体的体系，围绕消防、交通、网络及自然灾害等核心领域普及安全知识，引导选手运用编程与 AI 技术解决虚拟场景中的实际问题，并鼓励其开展应急方案创新设计，旨在打造集知识学习、技能训练、创新实践与成果展示于一体的综合平台，切实增强青少年的安全意识、自救互救能力与科技素养，为社会培养面向未来的安全人才。

二、 参赛范围

1. 参赛组别：

小学低年级组（1-3 年级）、小学高年级组（4-6 年级）、初中组、高中组（含中专、职高）

2. 参赛人数：1 人

3. 指导教师：1 人（可空缺）

组别确定：以地方教育行政主管部门（教委、教育厅、教育局）认定的选手所属学段为准。

三、比赛流程

1. 报名方式：参赛选手需统一于大赛官方网站进行在线免费报名，并真实、准确且完整地填写相关参赛信息。
2. 初赛：线上独立完成，以“知识答题 + 编程实践应用闯关 + 应急创新方案提交”为评比模式，按成绩排名晋级决赛。
3. 决赛：初赛晋级的选手按所在组别统一时间在线按顺序完成“知识进阶答题 + 编程实践应用闯关 + 应急创新方案提交”环节。
4. 监考说明：初赛、决赛全程启用“电脑主摄像头 + 移动设备副摄像头”双监控；比赛期间实时抓拍、录制监控画面及屏幕操作，确保选手独立完成所有任务，杜绝替考、协作等违规行为。

四、比赛内容

本竞赛聚焦应急安全知识、编程实践与创新方案设计的深度融合，以知识引领、技术赋能为导向，推动青少年在掌握安全常识的基础上，探索科技在应急管理中的价值，全面提升安全素养与逻辑实践能力。

（一）组织形式

采用全线上形式进行，选手需通过官方指定的在线竞赛平台参与所有环节。所有任务均在线完成并提交，系统自动记录答题、编程闯关结果；选手需遵守独立参赛要求，确保比赛规范性与公平性。

（二）核心规则

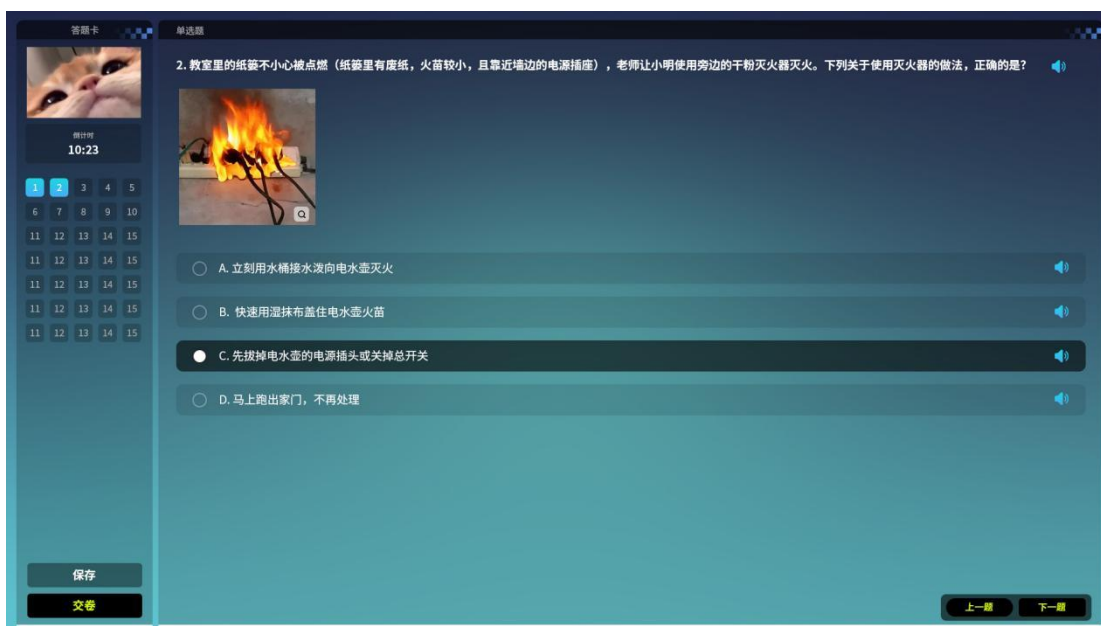
初、决赛均需选手在单场 60 分钟总时长内，在线依次完成应急知识答题、编程实践闯关，最后在规定时间内上传创新方案相关材料；初赛、决赛核心规则

框架一致，仅题目难度、创新方案具体要求有差异。

1. 应急知识答题：

均为客观题（单选 + 多选），以科普普及应急安全基础知识为核心；答题完成后由系统自动判分。

以下为应急知识答题样例。

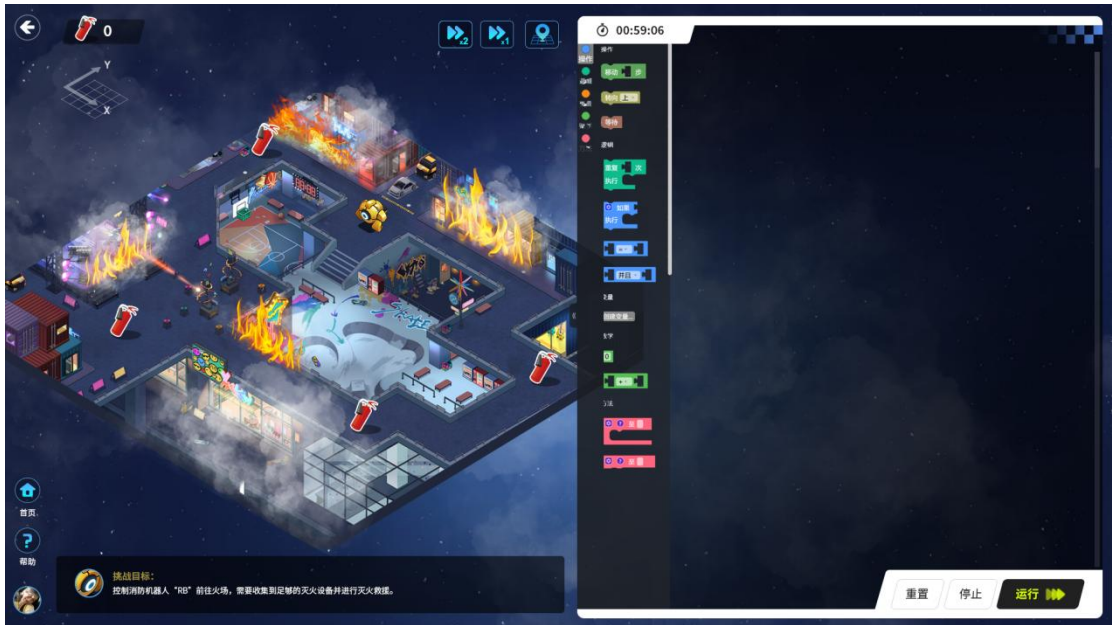


✦ 应急知识答题例题 ✦

2. 编程实践闯关：

以任务关卡形式呈现，每关单独计分且分数递增；选手需通过编程语言，在虚拟应急场景中编写程序完成消防、交通、网络安全、自然灾害等应急安全主题任务（如规划逃生路线、模拟人员疏散等），任务完成后由系统根据统一评分标准自动判分。

以下为编程实践关卡任务样例，实际场景和操作界面会有所调整。



✦ 编程实践应用例题

3. 创新方案提交:

提交应急创新方案设计; 主题侧重“应急场景痛点发现 + 简单解决方案设计”, 小学组可结合生活经验设计基础安全方案(如“家庭安全提醒清单”); 中学组可融入基础技术思路(如“基于编程的简易预警构想”); 高中组侧重方案落地简易性, 可适当补充技术验证或场景延伸思考。具体材料形式、内容要求等, 将在赛事手册中明确。

(三) 考察范围

聚焦三大核心维度, 内容深度适配不同学段能力水平。

- 1、应急安全知识普及:
- 2、通过答题覆盖消防安全、交通安全、网络安全、自然灾害应对等核心领域, 强化基础认知;
- 3、技术应用能力考察: 要求学生自主选择编程语言, 在虚拟应急场景中完

成程序编写与任务落地；

4、创新设计能力培养：侧重应急场景痛点挖掘与基础方案设计，进行方案细节优化与可行性提升，形成递进式能力考察体系。

五、评比标准

（一）计分说明

任务模块	环节	计分机制	满分
应急知识答题	初赛&决赛	按正确率计分，系统自动判分	40分
编程实践闯关	初赛&决赛	每关独立评价，得分按关卡数递增 (如 5分→10分→15分→...)	累积积分
创新方案设计	初赛	按考察维度评分	30分
创新方案设计	决赛	按考察维度评分	40分

（二）评分维度细则

1、编程实践闯关：每关独立评分，未完成不计分；

(1) 核心维度：

- 关卡完成度 (20%)：程序是否正确运行并达成基本的通关条件。
- 应急任务完成度 (40%)：是否完成关卡要求的应急安全任务目标。
- 代码简洁性与效率 (40%)：以官方标准答案为基准，综合考量代码/积木块的简洁度与逻辑效率。代码长度或积木块数量超出基准将按

比例扣分。

2、创新方案设计：

(1) 初赛评分依据：

- 痛点精准度（30%）：所关注的应急场景问题是否真实、明确。
- 方案逻辑性与创新性（40%）：解决方案的构思是否合理、清晰，并具有一定新意。
- 学段适配性（30%）：方案的复杂度和实现思路是否符合对应学段（小、初、高）的认知与实践能力。

(2) 决赛评分依据：

- 方案优化与深度（30%）：相较于初赛，方案细节是否更完善，思考是否更深入，可行性是否提升。
- 应用价值与创新性（40%）：方案是否具备实际应用潜力或启发价值，思路是否新颖。
- 呈现清晰度（30%）：提交的文档、PPT 等材料是否逻辑分明、重点突出、表达流畅。

（三）成绩计算

选手按总成绩（应急知识答题、编程实践闯关、创新方案设计三部分得分之和）进行排名，总成绩高者名次列前；总成绩相同者，比赛用时短者排名靠前；若仍相同，则列为并列名次。

六、编程语言

1. 图形化 Scratch：小学 1-3 年级、小学 4-6 年级

2. Python：小学 4-6 年级、初中、高中（含中专、职高）

3. C++：小学 4-6 年级、初中、高中（含中专、职高）

每位选手在报名时只能选择一种编程语言参赛。

七、比赛环境

1. 设备要求：参赛选手需自备符合要求的电脑设备，（台式机、笔记本均可，不可使用手机、Pad、平板电脑等移动设备参赛）。

2. 其他设备：电脑需配备内置或外接摄像头（用于身份核验与过程监控）及音频输入设备（部分情况需语音交互）。

3. 网络要求：活动期间需保证网络通畅（建议带宽 ≥ 50 Mbps，避免卡顿或断连）。

4. 浏览器要求：使用谷歌 Chrome 最新版本（避免兼容性问题）。

5. 监考要求：比赛全程开启摄像头及屏幕录制，另备一台移动设备用作第二机位监考。

八、不予评奖

凡出现以下情况之一的，选手将取消活动资格或不予评奖，成绩作废：

1. 找他人替考或替他人考试

2. 迟到 20 分钟以上（以竞赛系统登录时间为准）；

3. 选手最终成绩为零分（任一环节未提交成果）；

4. 选手被投诉存在违规行为（如抄袭方案、复制代码），经组委会核查属实；

5. 线上比赛期间，选手端监控画面出现除选手外的其他人员，或故意遮挡摄

像头、未拍摄关键操作区域；

6. 线上活动期间，选手端摄像头画面出现除选手外的其他人员（如家长、指导教师）。

九、相关说明

（一）参赛必须由选手独立完成，一旦发现有违规行为，主办方有权取消参赛资格。选手须保证知识答题答案、编程作品及创新方案均为原创，不触犯国家相关法律法规及政策，不涉及色情、暴力、歧视等违反道德规范的内容；若因著作权、肖像权、知识产权等产生纠纷，法律责任由选手本人承担，主办方保留取消选手参赛、获奖资格及追回奖项的权利。

（二）主办方拥有活动最终解释权。凡报名并提交作品者视为已取得本人及监护人同意，已知悉、确认并同意本次活动的各项规定。主办方对参赛的创新方案、编程作品拥有非商业性展示、推广、交流的权利（需注明作者姓名、学校及学段），不另付报酬。

（三）本规则是实施裁判工作的依据，在活动过程中裁判（评委）有最终裁定权。凡是规则中没有说明的事项由裁判组决定，未明确的事项由组委会通过官网公告补充说明。

GoC 编程挑战赛

评分规则

比赛以 C++ 语言为核心基底，创新融入冰雪运动、冰雪景观等元素，要求选手运用 C++ 语法与算法逻辑编写代码，实现指定的输出效果。

比赛给出六道试题，选手按题目要求，在代码区编写及调试程序代码，并可试运行，在展示效果区可以看到运行结果。选手努力在尽量短的时间内，提交代码答案，使运行结果达到题目要求。

一、题型及分数设置

- 小学低年级组

年级：小学二、三、四年级

题型：完善程序题 2 题，编程题 4 题，50 分/题，共 300 分

时长：60 分钟

- 小学高年级组

年级：小学五、六年级组

题型：编程题 6 题，50 分/题，共 300 分

时长：60 分钟

- 中学组

年级：七年级、八年级、九年级

题型：编程题 6 题，50 分/题，共 300 分

时长：60 分钟

二、赛题样例

1. 完善程序题样例：

题 1：

一滴水珠（程序填空）

请编写程序，画出这滴水珠。水珠是由半径为0.1.2.3.4.5...89的90个圆组成，相邻两圆之间距离为：2，颜色代码：12。



作答区域

画坐标← 运行命令 ↑ ↓ ↵

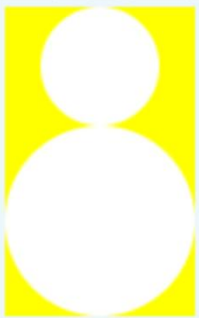
```
1 int main() {
2     p.up().hide();
3     for(int i = 0; i < 90; i++)
4         p.oo(□, 12).bk(2); // 循环变量作为实心圆的半径
5     return 0;
6 }
```

题 2：

简单小雪人（程序填空）

小Q住在北方，冬天到了，下雪啦，小Q堆了一个雪人。小Q十分喜欢自己的雪人，他担心大风会破坏雪人，想找一个大小刚好的箱子围着雪人，请你把箱子和雪人画出来。

雪人由两个圆心在同一直线上的圆构成，小圆在上面，大圆在下面，两个圆为白色（15号色）。矩形箱子是黄色（5号色）。小圆半径是：50；大圆半径：80；



作答区域

画坐标← 运行命令 ↑ ↓ ↵

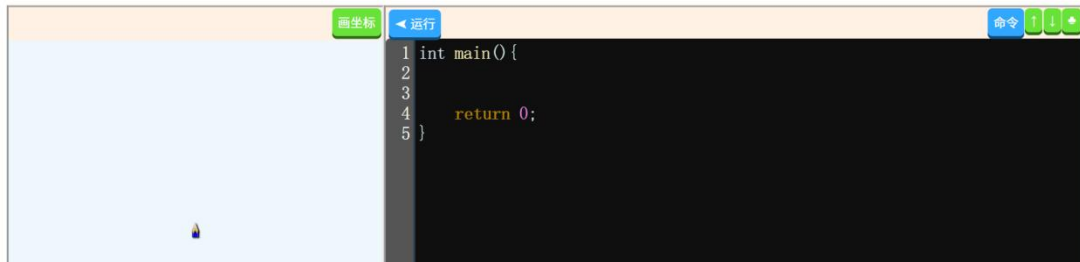
```
1 int main() {
2     p.up();
3
4     p.rr(□, □ + □, 5);
5     p.fd((□ + □) / □).bk(50/2.0);
6     p.oo(50/2.0, 15);
7     p.bk(50/2.0+80/2.0);
8     p.oo(80/2.0, 15);
9
10    p.hide();
11
12    return 0;
13 }
```

2. 编程题样例

题 1:

一滴水珠

请编写程序，画出这滴水珠。水珠是由半径为0.1.2.3.4.5...89的90个圆组成，相邻两圆之间距离为：2，颜色代码：12。

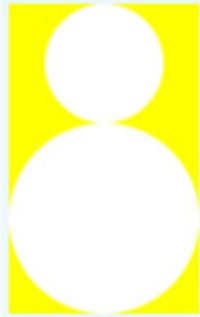


题 2:

简单小雪人

小Q住在北方，冬天到了，下雪啦，小Q堆了一个雪人。小Q十分喜欢自己的雪人，他担心大风会破坏雪人，想找一个大小刚好箱子围着雪人，请你把箱子和雪人画出来。

雪人由两个圆心在同一直线上的圆构成，小圆在上面，大圆在下面，两个圆为白色（15号色）。矩形箱子是黄色（5号色）。小圆半径是：50；大圆半径：80；



三、答题要求

选手按题目要求编辑代码并提交，代码运行结果需达成题目目标结果。答题结束后提交，由系统自动对代码评测打

分。每道题设若干个测试点，每通过一个测试点可得相应的分数。

答题界面如下图所示：

The screenshot displays a programming competition interface. At the top, a red-bordered box contains the problem description. The text reads: "小pen正在研究一种特殊的雪花:" followed by three blue snowflake diagrams. The first has 4 arms, the second has 5, and the third has 8. The text continues: "研究发现，它们都是由一些边长为50的小“叉”组成的，第1个图有4个小叉，第2个图有5个小叉，第3个图有8个小叉。小pen浪漫的想，不仅要有不同的小叉个数，还应该配上2种颜色间隔的画才好看。比如输入小叉个数n=6,两种颜色号a=3、b=4,画出的图如下:" followed by a green and cyan snowflake diagram. A note says: "注意：所有线段长度都一样为50." Below this is a "样例解释" section with the text "无".

The interface below the problem description is split into two main areas. On the left is the "结果展示区" (Result Display Area), which is currently empty. On the right is the "代码编辑区" (Code Editor Area), which contains a simple C++ program:

```
1 int main() {
2
3
4     return 0;
5 }
```

选手读题后，在“代码编辑区”编写和调试代码，“展示效果区”显示代码运行结果。

四、计分规则

选手提结果，系统自动评分。成绩排名考察两个参数：得分和用时。

(一) 首先以总得分的多少进行排名，总得分多的排名靠前。

(二) 若总得分相同，则按照总用时的长短排名，总用

时少的排名靠前。总用时为每个得分的问题所用时间之和。

(三) 每个题目都可以多次提交答案, 取其中最高得分作为该题的得分。该题所用的时间是从竞赛开始到该题最高得分提交的时间加上该问题的罚时(每道题提交超过一次的, 每多提交一次, 罚时 5 分钟)。得 0 分的问题不记时。

例如: A、B 两人都正确完成两道题目, 总得分相同, 其中 A 提交这两题的时间分别是比赛开始后第 10 分钟和第 15 分钟, B 的提交这两题的时间分别为第 11 分钟和第 13 分钟, 但 B 有一题提交了 2 次而产生罚时 5 分钟。这样 A 的总用时为 $10+15=25$, 而 B 总用时为 $11+13+5=29$, 所以 A 以总用时少而排名靠前。

(四) 如果选手间总得分和总时间都相同(此情形基本不会出现), 则比较第六题的得分和时间。如果仍都相同, 则比较第五题的得分和时间, 依次向前类推。

(注: 6 道题目的设置, 是按照由易到难的顺序, 最后一题难度最高)