

团 体 标 准

T/CIEXXX-2025

青少年机器人技术等级评价指南

Evaluation Guideline for Robot Technology Grade of Juniors

(草案)

2025 - XX - XX 发布

2025 - XX - XX 实施

中国电子学会 发 布

目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 青少年机器人技术等级评价指南概述	2
5 各等级知识点及评价标准	3
6 成绩评定	8

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件由中国电子学会提出。

本文件由中国电子学会科学普及工作委员会归口。

本文件起草单位：

本文件主要起草人：

青少年机器人技术等级评价指南

1 范围

本文件规定了青少年机器人技术的等级、器材使用、能力要求、评价标准及评价方法。

本文件适用于青少年机器人技术等级评价所涉及的个人、组织、机构或企业。本文件不适用于以学历教育或课程教学为目的的评价活动。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 29802 信息技术 学习、教育和培训测试试题信息模型

GB 1526-1989 信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定

GB/T 33265 教育机器人安全要求

3 术语和定义

GB/T 29802、GB/T 33265 界定的下列术语和定义适用于本文件。

3.1

青少年 Juniors

年龄在6岁到18岁之间的个体，此“青少年”约定仅适用于本文件，并建议8岁起参与。

3.2

机器人 Robot

由开源控制器、传感器和执行器及结构件组成，用于学习本文件所描述相关知识的器件组合，此“机器人”约定仅适用于本文件。

3.3

图形化编程平台 Visual Programming Platform

面向青少年学习计算机编程的程序编写平台。无需编写文本代码，只需要通过鼠标将具有特定功能的模块按照逻辑关系拼接起来就可以实现编程。

3.4 开源主控板 Open Source Main Control Board

主控芯片采用Atmega328P芯片，输入输出引脚编号与Arduino开源UNO/Nano控制板一致的主控板。开源主控板是青少年机器人技术等级三、四级指定用主控板。此“开源主控板”约定仅适用于本文件。

3.5

物联主控板 IOT Main Control Board

主控芯片采用ESP32-D0WD芯片，输入输出引脚编号与ESP32开源DevKitC控制板一致的主控板。物联主控板是青少年机器人技术等级五、六级指定主控板。此“物联主控板”约定仅适用于本文件。

3.6

智能主控板 Intelligent Main Control Board

RDK X3/X5主控板。此“智能主控板”约定仅适用于本文件。

3.7

C/C++语言 C/C++ Language

指以C/C++为基础的Arduino C/C++、ESP32 for Arduino编程语言。此“C/C++语言”约定仅适用于本文件。

3.8

舵机 Servo

指按照脉冲持续时间控制转动角度的执行器。此“舵机”约定仅适用于本文件。

3.9

了解 Know

对知识、概念或操作有基本的认知，能够记忆和复述所学的知识，能够区分不同概念之间的差别。

3.10

理解 Understand

在了解的基础上，能够明白事物背后的机制和原理，能够复现相关的操作，能对知识和技能进行简单的运用。

3.11

掌握 Master

在理解的基础上，能够根据类似的场景和问题，进行综合分析并正确运用所学知识和技能创造性的解决问题。

4 青少年机器人技术等级评价指南概述

4.1 目的

本文件从青少年制作机器人所需要的技能和知识出发，以规范化评价青少年机器人技术等级为中心，引导地区的组织、机构及企业根据当地机器人教育普及情况搭建适合当地的青少年机器人技术培养模式，从而激发和培养青少年学习现代机器人技术的热情和兴趣，引导青少年建立工程化、系统化的逻辑思维，提升力学、机械原理、电子信息和软件技术等学科的入门知识，培养学生的实践能力。

本文件不以评价教学为目的。

4.2 评价对象

本文件适用于3.1条所指定的青少年。

4.3 等级划分

青少年机器人技术等级共分为一级至八级共八个等级，八个等级之间呈现分级递进、螺旋上升的关系。文件中每个等级对应的能力标准如表1所示。文件中描述了每个等级相应的核心知识点对知识点的掌握程度要求。

表1 青少年机器人技术等级能力描述

等级	能力要求	能力描述
一级	机器人基本结构认知和搭建能力	认识日常工具和基本结构，能够合理使用三角形、杠杆、滑轮、轮轴、齿轮传动装置、链传动装置等搭建简单无动力结构，具备基本结构搭建的能力。
二级	机器人驱动与传动系统认知和搭建能力	熟悉机器人基本常识，能够合理使用直流电机、棘轮机构、连杆机构、凸轮机构、平面四杆机构、皮带传动装置等搭建动力驱动结构，具备复杂结构搭建能力。
三级	机器人基础控制能力	基于图形化编程平台，应用顺序、循环、选择三种基本结构，通过编程实现简单交互装置，实现简单软硬件协同，具备基础自动控制能力。
四级	机器人自动控制能力	基于C/C++代码编程，通过编程实现传感器数据读取、控制执行器运动，实现较复杂软硬件协同，具备自动控制能力。
五级	机器人通信交互能力	在四级基础上，通过编程实现中断控制、数据位读写操作、串口通信，具备基本的数据交互能力。
六级	机器人物联网控制能力	在五级基础上，通过编程利用I ² C和SPI进行串行通信，通过Wi-Fi连接控制和通过Web服务器进行数据交互，具备较完备的软硬协同闭环控制能力。
七级	机器人智能处理能力基础	基于Linux平台，使用Python语言编程，通过OpenCV进行图像处理，具备基于ROS的机器人智能信息处理的基础能力。
八级	机器人智能处理能力	掌握基本的数据结构和迷宫探索算法，掌握基于深度学习的图像数据采集、标注、训练和推理，能够利用AI工具，具备机器人智能信息处理能力和系统工程思维。

5 各等级知识点及评价标准

5.1 一级标准

5.1.1 考核知识点及知识点掌握程度要求

一级标准的知识点如表2所示，知识点排序不分先后。

表2 一级标准知识点

编号	知识点名称	知识点掌握要求
1	机器人常识	了解主流的机器人影视作品及机器人形象
2	基本工具和零件	认识日常生活中的基本工具和零件
3	斜面	熟悉斜面原理、特性、日常应用
4	基本结构	掌握稳定结构和不稳定结构的特性，掌握三角形、平行四边形特性及应用
5	杠杆	了解杠杆原理五要素，掌握省力杠杆、费力杠杆等臂杠杆的特点和应用，掌握杠杆的搭建
6	齿轮	了解齿轮的种类、掌握齿轮传动装置的分类、特性、搭建
7	轮轴	掌握轮轴的原理、特性、应用、搭建
8	滑轮	掌握动滑轮、定滑轮、滑轮组的原理、特性、应用、搭建
9	链传动	了解履带基本特点，掌握链传动装置的组成、特性、搭建
10	其他	单摆、传动比的简单计算、风扇降温原理

5.1.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，机械结构搭建采用实际操作的形式。

5.1.3 器材

能够满足实际操作考试要求的结构组件均可。

5.2 二级标准

5.2.1 考核知识点及知识点掌握程度要求

二级标准的知识点如表3所示，知识点排序不分先后。

表 3 二级标准知识点

编号	知识点名称	知识点掌握要求
1	机器人常识	了解中国及世界机器人领域的重要历史事件及重要理论
2	摩擦力	了解摩擦力分类、产生条件、应用
3	前驱和后驱	了解前驱和后驱的基本特性
4	棘轮机构	掌握棘轮机构的组成、动力传递方式、分类、应用、搭建
5	曲柄机构	掌握曲柄机构的组成、动力传递方式、应用、搭建
6	皮带传动	掌握皮带传动装置的组成、分类、特性、搭建
7	凸轮机构	掌握凸轮机构的组成、动力传递方式、分类、应用、搭建
8	连杆机构	掌握连杆机构的组成、特性
9	平面四杆机构	掌握平面四杆机构的概念、分类、动力传递方式、应用、搭建
10	间歇运动机构	了解常见的间歇运动机构
11	其他	电动机的能量转化、飞机升降原理、伯努利定理、重力方向、差速运动控制、行星齿轮、挖掘机及老式缝纫机的基本构成等

5.2.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，机械结构搭建采用实际操作的形式。

5.2.3 器材

能够满足实际操作考试要求的结构组件均可。

5.3 三级标准

5.3.1 考核知识点及知识点掌握程度要求

三级标准的知识点如表4所示，知识点排序不分先后。

表 4 三级标准知识点

编号	知识点名称	知识点掌握要求
1	机器人常识	了解机器人领域的相关理论、相关人物及前沿科技时事
2	主控板	了解开源主控板的基本性能
3	基本电路	理解串联电路、并联电路的基本特性，掌握串联电路、并联电路的搭设
4	导电材料	了解导体、半导体、绝缘体的基本特性及常用分类
5	欧姆定律	理解电流、电压、电阻的概念及三者间的相互关系
6	图形化编程平台使用	掌握图形化编程平台使用，能够进行程序的编写、调试、上传
7	信息处理基本流程	理解“输入、处理、输出”信息处理基本流程，掌握通过图形化编程，实现简单交互程序的编写
8	基本编程技能	图形化编程环境下，掌握程序设计顺序、选择、循环三种基本结构、变量的定义、数学运算符、比较运算符、逻辑运算符的使用
9	数字信号	理解数字信号的基本概念，掌握图形化编程环境下数字信号的读写
10	模拟信号	理解模拟信号的基本概念，掌握图形化编程环境下模拟信号的读写
11	流程图	掌握程序流程图的绘制

12	分立器件	了解LED、按键开关、光敏电阻、电位器等常见分立器件的基本工作原理，掌握通过图形化编程实现数据读写操作
13	传感器模块	了解超声波传感器、红外遥控传感器的基本工作原理，掌握通过图形化编程实现数据读取操作
14	执行器模块	了解舵机的基本工作原理，掌握通过图形化编程实现数据写入操作

5.3.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建及程序编写采用实际操作的形式。

5.3.3 编程语言及器材

理论知识考试试卷中编程题采用图形化编程平台Mixly软件编写，实际操作考生不指定具体的图形化编程平台。

器材为采用开源主控板并能够满足实际操作考试要求的组件均可。

5.4 四级标准

5.4.1 考核知识点及知识点掌握程度要求

四级标准的知识点如表5所示，知识点排序不分先后。

表 5 四级标准知识点

编号	知识点名称	知识点掌握要求
1	机器人常识	了解机器人领域的相关理论、相关人物及前沿科技时事
2	主控板	理解开源主控板的基本性能
3	数制	掌握整数在二进制、八进制、十进制和十六进制之间进行转换
4	基本编程技能	使用C/C++语言编程，掌握程序设计顺序、选择、循环三种基本结构，掌握变量的定义、变量的作用域，掌握数学运算符、比较运算符、逻辑运算符的使用
5	数字信号	掌握数字信号的基本概念，掌握高低电平、上拉电阻电路、下拉电阻电路的基本概念，掌握采用代码编程实现数字信号的读写
6	模拟信号	掌握模数转换的基本原理，掌握PWM模拟输出的基本原理，掌握采用代码编程实现模拟信号的读写
7	类库	理解类库的概念，掌握类库的安装及类库成员函数的调用
8	传感器模块	理解灰度传感器、按键模块、触碰传感器、超声波传感器、红外遥控传感器的基本工作原理，掌握通过代码编程实现数据读取操作
9	执行器模块	理解舵机、直流电机马达驱动模块的基本工作原理，掌握通过代码编程实现对执行器的运动控制
10	三极管	了解三极管的基本特性，掌握通过代码编程，通过三极管控制电路通断
11	机器人控制	理解开环控制和闭环控制的基本概念，掌握简单开环和闭环机器人制作

5.4.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建及程序编写采用实际操作的形式。

5.4.3 编程语言及器材

编程语言采用C/C++。

器材为采用开源主控板并能够满足实际操作考试要求的组件均可。

5.5 五级标准

5.5.1 考核知识点及知识点掌握程度要求

五级标准的知识点如表6所示，知识点排序不分先后。

表 6 五级标准知识点

编号	知识点名称	知识点掌握程度要求
1	机器人常识	了解机器人、微控制器领域的相关理论、相关人物及前沿科技时事
2	主控板	理解物联主控板的基本性能，掌握利用物联主控板进行数字信号、模拟信号读写操作，掌握板载电容触摸传感器、霍尔传感器的数据读取操作
3	传感器模块	理解热释电模块、编码器模块的基本工作原理，掌握通过编程实现数据读取操作
4	执行器模块	理解一位数码管、四位数码管、8x8点阵的工作原理
5	中断	理解中断的运行机理，掌握中断回调函数的使用
6	数组	掌握一维数组、二位数组的应用
7	位操作	掌握数据位的操作
8	UART串行通信	理解UART串行通信基本工作原理，理解报文的含义和组成，掌握利用串口Serial类库进行串口数据的读写操作
9	字符串	掌握利用字符串String类库对字符串进行解析处理
10	按键消抖	掌握通过软件实现按键消抖
11	移位寄存器芯片	理解移位寄存器芯片74HC595的基本工作原理，掌握通过移位寄存器芯片74HC595进行一位数码管、四位数码管、8x8点阵的显示控制
12	E ² PROM	理解E ² PROM的基本工作原理，掌握利用E ² PROM类库进行数据读写
13	蓝牙通信	理解经典蓝牙通信的基本工作原理，掌握通过蓝牙进行数据接收、发送

5.5.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建及程序编写采用实际操作的形式。

5.5.3 编程语言及器材

编程语言采用C/C++。

器材为采用物联主控板并能够满足实际操作考试要求的组件均可。

5.6 六级标准

5.6.1 考核知识点及知识点掌握程度要求

六级标准的知识点如表7所示，知识点排序不分先后。

表7 六级标准知识点

编号	知识点名称	知识点掌握程度要求
1	机器人常识	了解机器人、微控制器领域的相关理论、相关人物及前沿科技时事
2	I ² C串行通信	理解I ² C串行通信基本工作原理，掌握通过类库进行数据交互
3	SPI串行通信	理解SPI串行通信基本工作原理，掌握通过类库进行数据交互
4	姿态传感器	理解姿态传感器MPU6050的基本工作原理，掌握通过类库进行数据交互
5	激光测距传感器	理解激光测距传感器VL53L0X的基本工作原理，掌握通过类库进行数据交互
6	液晶显示屏	掌握通过类库对SSD1306 OLED显示屏进行操作
7	互联网基础	了解TCP/IP协议、IP地址、端口、URL基础知识，理解HTML文档基本结构，理解物联网
8	Wi-Fi	理解WiFi类库，掌握通过WiFi类库以STA、AP模式实现Wi-Fi连接
9	Web服务器	掌握利用WiFi类库实现Web服务器的建立和数据交互
10	物联网	了解物联网的基础知识，理解MQTT协议的工作原理及基本应用
11	步进电机	理解步进电机的基本工作原理，掌握通过类库实现步进电机运动控制
12	机器人控制	理解PID控制器的基本工作原理，掌握使用编码器电机，通过比例控制，实现机器人按照指定线路运动

5.6.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建及程序编写采用实际操作的形式。

5.6.3 编程语言及器材

编程语言采用C/C++。
器材为采用物联主控板并能够满足实际操作考试要求的组件均可。

5.7 七级标准

5.7.1 考核知识点及知识点掌握程度要求

七级标准的知识点如表8所示，知识点排序不分先后。

表 8 七级标准知识点

编号	知识点名称	知识点掌握程度要求
1	人工智能常识	了解人工智能的发展过程，了解机器学习、神经网络、深度学习之间的相互关系及基本知识
2	Python语言基本编程技能	掌握Python基本语法，掌握基本数据类型及相关数据操作，掌握分支结构、循环结构的使用，掌握常用内置函数的使用，掌握异常处理程序的编写，掌握函数的定义及调用，掌握类的定义和使用，掌握文件的读取和写入，掌握常用标准库（math、random、time）的使用，掌握第三方库（NumPy、Matplotlib）的安装及基本使用
3	OpenCV图像处理	理解图像的基础知识，掌握图像及视频流的读取、保存、显示，掌握图像的阈值处理、几何变换、滤波等图像处理方法，掌握边缘检测、霍夫变换等特征提取方法，掌握使用深度学习模型进行图像特性的提取和识别
4	Linux基础	了解Linux的目录结构，掌握Linux基本的用户管理、文件和目录管理、权限管理、网络操作、包管理等基本命令
5	ROS	掌握ROS节点、功能包、工作空间、通信机制等基本概念及相关的操作命令，掌握节点间话题、服务、参数等通信机制的程序实现
6	智能主控板	了解智能主板板的基本性能，理解智能主控板的基本性能，掌握利用智能主控板进行数字信号、模拟信号读写操作，掌握智能主控板和外部器件进行数据通信，掌握智能主控板基于深度学习的多媒体应用

5.7.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建和程序编写采用实际操作的形式。

5.7.3 编程语言及器材

编程语言采用Python。
器材为采用智能主控板并能够满足实际操作考试要求的组件均可。

5.8 八级标准

5.8.1 考核知识点及知识点掌握程度要求

八级标准的知识点如表9所示，知识点排序不分先后。

表 9 八级标准知识点

编号	知识点名称	知识点掌握程度要求
1	数据结构和算法	理解算法复杂度的概念，掌握数组、链表、栈、队列、树、图等基本数据结构的数据存储、遍历，掌握基础的排序算法、搜索算法和路径规划算法。理解基础的机器学习算法
2	机器人自主运动	理解差速运动学原理，掌握运动控制器和智能主控板的信息交互，掌握基于规则的迷宫导航，掌握基于ROS和深度学习实现小车的自主运动
3	人工智能基础知识	掌握基于深度学习的图像数据采集、标注、训练和推理

5.8.2 考试形式

考试分为理论知识考试和实际操作考试。理论考试采用上机考试的形式，装置搭建及程序编写采用实际操作的形式。

5.8.3 编程语言及器材

编程语言采用Python和C/C++。

器材为采用智能主控板并能够满足实际操作考试要求的组件均可。

6 成绩评定

评价结果采用百分制计分。

成绩评定根据预设的、细化的评分标准进行打分，考生成绩达到规定的合格分数线（通常为60分）方视为通过该等级考核。

理论知识考试题型包括但不限于单项选择题、多项选择题、判断题等。

实际操作考试以是否实现考题的任务要求作为最终的评判标准。

成绩评定应由具备资质的考官或监考系统进行监督，确保考试的公平、公正。

团 体 标 准

T/CIE XXX-2025

青少年软件编程等级评价指南

第 1 部分：通则

Evaluation Guideline for Software Programming Grade of Juniors

Part 1: Guideline
(草案)

2025-XX-XX 发布

2025-XX-XX 实施

中国电子学会 发布

目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 评价总则	2
5 评价原则	3

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是T/CIE**《青少年软件编程等级评价指南》的第1部分，T/CIE**已发布了以下几个部分。

—第1部分：通则

—第2部分：图形化编程

—第3部分：C/C++语言编程

—第4部分：Python语言编程

本文件由中国电子学会提出。

本文件由中国电子学会科学普及工作委员会归口。

本文件起草单位：

本文件主要起草人：

青少年软件编程等级评价指南

第 1 部分：通则

7 范围

本文件规定了软件编程（涵盖图形化编程、C/C++语言编程及 Python 语言编程）的基本术语与定义、等级划分总则及评价原则。

本文件适用于相关组织、机构和企业开展青少年软件编程技术等级评价，但不适用于学历教育或课程教学范畴的评价活动。

8 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 5271.1 信息技术 词汇 第 1 部分：基本术语

GB/T 5271.7 信息技术 词汇 第 7 部分：计算机编程

GB/T 29802 信息技术 学习、教育和培训 测试试题信息模型

9 术语和定义

GB/T 5271.1、GB/T 5271.7、GB/T 29802界定的下列术语和定义适用于本文件。

9.1

青少年 Juniors

年龄在 6 岁到 18 岁之间的个体。

注：此“青少年”约定仅适用于本文件。

9.2

图形化编程 Visual Programming

一种无须编写文本代码，只需要通过鼠标拖曳相应的图形化符号，按照一定的逻辑关系完成拼接就能实现编程的形式。

9.3

Python 语言 Python Language

一种解释性、跨平台、面向对象的动态类型编程语言，其设计哲学是强调代码的可读性与简洁性，其典型特点是免费开源、具有丰富的扩展库。Python 语言有 2.x 和 3.x 两个版本，本标准指 3.x 版本。此约定仅适用于本文件。

9.4

C/C++语言 C/C++ language

C 语言是一种面向过程的高级通用编程语言，最初由 Dennis Ritchie 为了编写 Unix 而开发。C++ 语言是一种面向对象的高级通用编程语言，最初由 Bjarne Stroustrup 作为 C 语言的扩展而开发。由于 C++ 高度兼容 C，所以我们将这两种语言统称为 C/C++ 语言。

9.5

了解 know

对知识、概念或操作有基本的认知，能够记忆和复述所学的知识，能够区分不同概念之间的差别。

9.6

理解 understand

在了解的基础上，能够明白事物背后的机制和原理，能够复现相关的操作，能够对知识和技能进行简单的运用。

9.7

掌握 master

在理解的基础上，能够根据类似的场景和问题，进行综合分析并正确运用所学知识和技能创造性地解决问题。

10 评价总则

10.1 等级划分

青少年软件编程等级评价的划分，遵循“激发兴趣、培养思维、循序渐进、重在实践”的理念。评价体系旨在衡量青少年在计算思维、算法设计与问题解决等方面的能力成长，而非单纯考察知识点的记忆。

10.2 编程形式

本文件涉及的编程形式包括图形化编程与文本编程（如 C/C++、Python 等）。不同编程形式旨在适应不同年龄段和认知水平的青少年，为其提供适宜的入门与进阶路径。

10.3 形式与语言关系

本文件所涉及的编程形式及编程语言之间无预设的等级递进关系。图形化编程并非文本编程的必然前置基础，C/C++语言与 Python 语言亦无绝对的难易或高级之分。

注：基于不同编程形式及语言的特点，为降低学习门槛，建议可从图形化编程入门；在文本编程学习中，可考虑从语法相对简洁的 Python 语言开始。

10.4 能力层级关系

本文件对青少年软件编程能力总体上划分为四个递进层次，各层次的核心能力目标如下：

第一层次（基础认知与操作）：掌握基本的程序结构（如顺序、分支、循环），能够理解并运行简单程序，完成基础的交互操作。

第二层次（数据处理与基础算法）：掌握基本的数据类型和结构，能对数据进行简单处理，并运用基础算法（如枚举、排序）解决实际问题。

第三层次（抽象与模块化）：掌握函数、类、对象等概念，具备代码复用和模块化设计的能力，能编写结构更清晰、易于维护的程序。

第四层次（复杂问题解决与综合应用）：掌握进阶算法（如递归、动态规划）和人工智能等领域的初步应用，能够综合运用所学知识设计解决方案，应对较为复杂的场景。

10.5 等级与能力映射

不同编程形式及语言的具体等级设置，根据其技术特性和学习曲线，与上述能力层级建立对应关系，详见下表：

编程形式/ 语言	对应等级	对应能力层次	能力描述与定位
图形化编程	一级至二级	第一层次	编程入门，培养逻辑思维。主要面向小学阶段学生，通过拖拽积木完成简单动画、游戏和故事创作。
	三级至四级	第二、三层次	初步接触数据处理和简单算法，开始使用函数等概念进行模块化设计。
C/C++语言编程	一级至二级	第一层次	掌握基本语法和程序结构。定位小学高年级及初高中学生。
	三级至五级	第二层次	深入理解数据结构（如数组、字符串），掌握基础算法。
	六级至七级	第三层次	掌握函数、结构体/类，理解面向过程/面向对象编程思想。
	八级至十级	第四层次	掌握指针、高级数据结构（如链表、树）、复杂算法，并能解决综合性问题。
Python 语言编程	一级至二级	第一层次	掌握基本语法和程序结构。定位小学三年级以上及初高中学生。
	三级	第二层次	掌握列表、字典等数据结构，运用基础算法解决问题。
	四级至五级	第三层次	掌握函数、文件操作、异常处理，初步理解面向对象编程。
	六级	第四层次	掌握模块化开发、常用库（如数据分析、人工智能库）的应用，解决实际项目问题。

11 评价原则

11.1 能力为本

本文件不以评价教学为目的，重点考察青少年在软件编程方面的实践应用能力、计算思维和问题解决能力。考核内容设计应贴近现实应用场景。

11.2 科学规范

评价体系、试题设计和评分标准应科学、规范、统一。试题需符合 GB/T 29802（测试试题信息模型）等相关标准，确保评价过程的严谨性和评价结果的有效性。

11.3 题型综合

评价题型应兼顾客观性与主观性，全面评估考生的知识掌握与技能运用水平。包括：
 客观题（如选择题、判断题）：用于考察对基本概念、语法和原理的理解。
 主观题（如编程题、项目设计题）：用于考察算法设计、代码实现和项目构建的综合能力。

11.4 开放便捷

评价过程宜采用上机在线形式进行，在保证考试安全性的前提下，利用技术手段实现考试地点的灵活设置，降低参与门槛，提升评价效率。

11.5 发展导向

本文件旨在通过规范化的等级评价，为各组织、机构和企业提供能力参照基准，引导其结合本地实际情况，构建科学、合理的青少年编程人才培养模式，最终激发青少年学习兴趣，促进编程教育的健康发展。

团 体 标 准

T/CIE XXX-XXX

青少年软件编程等级评价指南 第 2 部分：图形化编程

Evaluation Guideline for Software Programming Grade of Juniors—
Part 2: Visual Programming
(草案)

2025 – XX – XX 发布

2025 – XX – XX 实施

目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 图形化编程等级评价指南概述	4
5 青少年图形化编程等级考核要求	5
6 评价方法	13
参考文献	1

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是T/CIE**《青少年软件编程等级评价指南》的第2部分，T/CIE**已发布了以下几个部分。

—第1部分：通则

—第2部分：图形化编程

—第3部分：C/C++语言编程

—第4部分：Python语言编程

本文件由中国电子学会提出。

本文件由中国电子学会科学普及工作委员会归口。

本文件起草单位：

本文件主要起草人：

青少年软件编程等级评价指南

第2部分：图形化编程

12 范围

本文件规定了青少年图形化编程的等级、能力要求、评价标准及评价方法。

本文件适用于青少年图形化编程等级评价所涉及的组织、机构或企业。本文件不适用于以学历教育或课程教学为目的的评价活动。

13 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 29802 信息技术 学习、教育和培训测试试题信息模型

GB/T 5271.13 信息技术 词汇 第13部分：计算机图形

GB 1526-1989 信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定

T/CIE ***-***青少年软件编程等级评价指南 第1部分：通则

14 术语和定义

下列术语和定义适用于本文件

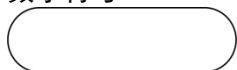
14.1

图形化编程基本符号

图形化编程形式是编程入门的主要手段，广泛用于基础编程知识教学及进行简单编程应用的场景。广义上来说采用图形形式进行编程的操作均可称为图形化编程形式，因此图形化编程形式多样。而本文件只是设定了一种图形化编程的形式，并针对这个图形化编程形式给出了一些指导性原则，遵循这些原则可以规范图形化编程的形式。图形化编程中包含具有确定含义的符号和简单的说明性文字，基本符号不涉及说明性文字的内容，不过每个符号均有一个无歧义且有意义的名称，它在整个文件编制中是一致的。

14.1.1

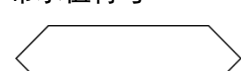
数字符号



此符号表示数字数据、文本数据或返回值为数字或文本的指令块。如果数据为变量则在符号中标明变量名，如果为指令块则符号当中应包含文字说明指令块执行的功能，如果有必要还可以包含表示参数的数据符号和布尔值符号。

14.1.2

布尔值符号



此符号表示布尔型的值或返回值为布尔值的指令块。如果数据为变量则在符号中标明变量名，如果为指令块则符号当中应包含文字说明指令块执行的功能，如果有必要还可以包含表示参数的数据符号和布尔值符号。

14.1.3

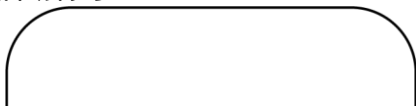
处理符号



此符号表示无返回值的指令块。符号当中应包含文字说明指令块执行的功能，如果有必要还可以包含表示参数的数据符号和布尔值符号。

14.1.4

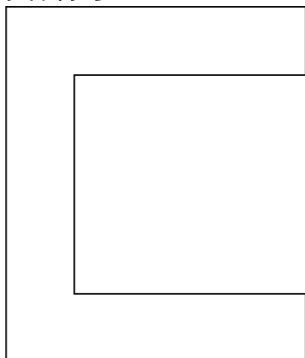
启动符号



此符号表示启动下方所连接指令块运行的条件。符号当中应包含文字说明启动程序的条件，如果有必要还可以包含表示参数的数据符号和布尔值符号。

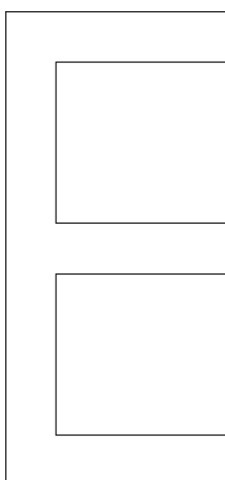
14.1.5

条件符号



此符号表示满足条件即可执行其中包含的指令块，符号顶部当中应包含必要的说明文字，以及表示条件的布尔值符号。

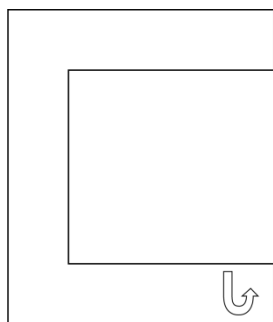
如果当条件为“真”和“假”时均有需要执行的指令块时，则符号可表示为



其中上边的凹槽中放置条件为“真”时执行的指令块，下边的凹槽中放置条件为“假”时执行的指令块。

14.1.6

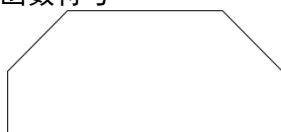
循环符号



此符号表示满足条件即不断重复执行其中包含的指令块，符号顶部当中应包含必要的说明文字，以及表示条件的布尔值符号。

14.1.7

函数符号



此符号表示其下方所连接的指令块能够组成一个函数。符号当中应包含文字表示函数名称以及必要的说明文字，如果有必要还可以包含表示参数的数据符号和布尔值符号。

14.1.8

图形化编程符号内的说明文字

本文件中所有的符号都要求在符号内列出说明文字。要把理解某个符号的功能所需的最低限度的说明文字置于符号内，文字应按照从左到右的方式来书写。

14.1.9

符号的宽度

符号的宽度根据实际情况可以不同。

14.1.10

符号的连接

同一级别的符号要求上下连接，连接时要左对齐。

本文件中的符号连接处均为水平的直线，而在具体的图形化编程环境中，上下连接的符号之间可以有一些缺口状的设计。

14.2

图形化编程平台 visual programming platform

提供图形化编程符号用于完成程序编写的软件平台。在软件平台中无需逐字逐句的编写代码，只需用鼠标拖曳的形式将正确的图形化编程符号拼接在一起（必要情况下可通过实体键盘或虚拟键盘输入参数）即可编写一段程序。

本文件主要要求学习图形化编程的青少年能够制作一个多媒体形式的作品，因此规定图形化编程环境中必须有能够展示作品的窗口，同时要能够通过图形化编程符号显示或隐藏图像以及控制播放音频。另外要求图形化编程环境中能够完成作品的保存与打开。

14.3

指令积木 program block

图形化编程环境中基于图形化编程符号定义的程序积木或指令块。

14.4

舞台 stage

图形化编程平台中展示多媒体作品的窗口。本文件规定窗口中需要包含控制程序运行和停止的按钮。此“舞台”约定仅适用于本文件。

在窗口中可以使用平面的坐标系来表述各种元素在窗口中的位置，该平面坐标系的原点在窗口中间，水平方向为x轴，向右为正，向左为负。垂直方向为y轴，向上为正，向下为负。

14.5

背景 background

舞台窗口中最底层的图像。通常作为多媒体作品展示的场景信息。建议在图形化编程环境中提供相关素材供使用者使用。

注：此“背景”约定仅适用于本文件。

14.6

角色 spirit

窗口中各种元素的统称。通常一个角色可以包含图像、声音以及代码的信息。

注：此“角色”约定仅适用于本文件。

14.7

造型 costume

角色拥有的不同样式的图像信息。建议在图形化编程环境中提供相关图像供使用者使用。

注：此“造型”约定仅适用于本文件。

14.8

脚本 script

针对角色和背景编写的由指令积木构成的图形化代码。

注：此“脚本”约定仅适用于本文件。

14.9

画笔 pen

角色具有的一种功能，利用该功能可以在舞台区绘制出角色运动的轨迹。画笔的颜色与粗细通过指令积木可以设置。

注：此“画笔”约定仅适用于本文件。

14.10

列表 list

列表可以看成是许多变量的有序集合，这个有序的集合应能够在舞台区显示出来。

注：此“列表”约定仅适用于本文件。

15 图形化编程等级评价指南概述

15.1 评价目的

本文件从软件编程所需要的技能和知识出发，以规范化评价青少年图形化编程为中心，引导地区的组织、机构及企业根据当地图形化教育普及情况搭建适合当地的青少年图形化编程培养模式，从而激发和培养青少年学习编程技术的热情和兴趣。让考生能够掌握图形化编程的相关知识和操作能力，熟悉编程各项基础知识和理论框架，通过设定不同等级的考试目标，让考生具备图形化编程能力，为后期专业化编程学习打下良好基础。

本文件不以评价教学为目的，考核内容不是按照学校要求设定，而是根据图形化编程所需能力以及国内图形化编程教育的普及情况而定，主要以实践应用能力为主。

本文件不限定图形化编程环境的具体软件，基于本文件开发的图形化编程环境均可作为参与等级评价的软件。针对于具体的图形化编程环境，可以给出对应于图形化编程环境的考试大纲与说明。

15.2 评价对象

本文件适用于6岁-18岁青少年。

15.3 等级划分

图形化编程等级共分为四个等级，对应等级能力依次从编程工具的基础操作到具备模块化编程与算法思维的能力。级别之间的逻辑关系及等级对应的能力描述如表1和表2所示，文件第五章中描述了每个等级相应的核心知识点和对知识点的掌握程度要求。

四个等级之间呈现由浅入深、能力递进的关系：

一级、二级为基础阶段，重点在于熟悉编程环境、掌握程序基本结构和实现简单的程序交互。

三级、四级为提高阶段，引入变量、函数、列表和算法，培养逻辑思维和程序结构化能力。

每个等级的知识与技能均建立在前一等级的基础上，形成从操作到思维、从单一到综合的完整能力链条。

由于在 Python 或 C/C++编程等级评价中依然会涉及到函数与算法的相关内容，因此建议达到三级之后可以参与 Python 或 C/C++编程等级评价。

表1 图形化编程等级逻辑关系

级别	逻辑点
一级→二级	从顺序结构过渡到包含分支与循环的结构，引入侦测与画笔功能，增强程序的逻辑与交互能力。
二级→三级	在程序基础结构中增加变量与逻辑运算，引入随机数与数学运算，培养数据意识与复杂逻辑处理能力。
三级→四级	从复杂逻辑过渡到列表、函数与算法的操作，引入模块化编程思想，提升程序结构与问题分解能力。

15.4 能力要求

表2 图形化编程等级能力要求

等级	能力描述	核心知识点
一级	初步学会使用编程工具，并按照要求编写出完整的顺序结构程序	掌握编程环境的基本操作 掌握拖拽指令积木的操作 理解参数的概念 了解舞台、背景、角色、造型的概念 完成顺序结构的程序 掌握顺序结构流程图的画法
二级	理解程序的三种基本结构，能够按照要求编写出能够进行交互的程序	理解坐标系的概念 掌握侦测与画笔的功能 完成包含分支与循环结构的程序 掌握数据的逻辑运算与关系运算 掌握选择结构、循环结构的流程图画法
三级	理解变量的概念，能够创建变量并处理变量中的数据	理解随机数、变量，以及变量作用域的概念 掌握逻辑运算与关系运算的组合使用 掌握选择语句、循环语句的嵌套使用 完成能够进行数据处理的程序
四级	理解列表和函数的概念，能够利用函数优化程序，能够通过函数学习算法知识	掌握对列表的操作 理解函数的作用范围 利用函数实现模块化编程 了解多线程的概念 理解排序、递推、递归、分治等常见算法

16 青少年图形化编程等级考核要求

16.1 一级评价要求

16.1.1 评价目标与能力要求

了解编程环境的界面，掌握编程环境的基本操作，能够导入角色造型和设置背景，并通过对角色的不同操作以及加入声音形成一个具有简单顺序结构的作品。考查简单的逻辑推理能力。

具体包括 2 方面能力要求：

1.初步学会使用图形化编程工具，理解编程环境中的核心概念

- (1) 理解编程环境界面中功能区的分布与作用；
- (2) 掌握拖拽指令积木到程序区的操作并进行正确的连接；
- (3) 掌握通过舞台区按钮完成运行与停止程序的操作；
- (4) 掌握角色的移动、旋转指令积木的使用；
- (5) 掌握如何设置声音的播放；
- (6) 掌握如何进行背景切换；
- (7) 掌握计算机上已保存作品的打开以及保存新制作的作品。

2.按照规定的要求编写出完整的顺序结构程序

- (1) 掌握顺序结构流程图的画法；
- (2) 理解参数的概念，能够调整指令积木中的参数；
- (3) 掌握如何完成一个顺序结构的程序；
- (4) 掌握在程序中包含播放一段音频和切换背景的操作；
- (5) 掌握程序中包含切换角色的造型，角色移动和旋转的操作。

16.1.2 包含的指令积木

本文件中只约定了最少要包含的指令积木（下同）。

当点击舞台区的启动程序按钮

当点击启动按钮时

角色在舞台中的移动

移动 像素

角色在舞台中的角度（0 度为正上方）

转到 度

角色旋转

左转 度

右转 度


播放声音

播放声音


切换背景

将背景切换为

角色切换造型

将造型切换为 

等待

等待  秒

角色显示文本内容

说   秒

下一个造型

下一个造型

下一个背景

下一个背景

16.2 二级评价要求

16.2.1 评价目标与能力要求

深入理解图形化编程环境的操作，能够设置多个角色的位置以及上下层的关系，可以通过交互和选择指令积木的综合应用解决问题考查对循环语句的掌握程度。考查一般逻辑推理和总结归纳能力。

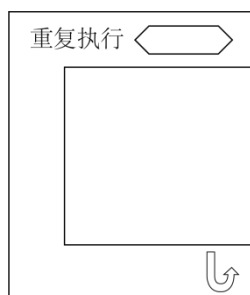
具体包括以下2方面能力要求：

1. 掌握图形化编程环境的使用

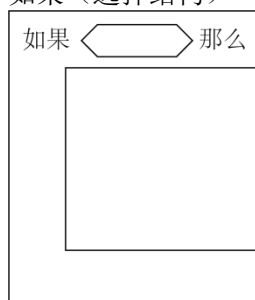
- (1)理解舞台区中层的概念；
 - (2)理解舞台区坐标系的概念；
 - (3)掌握利用随机位置指令积木让角色移动到随机位置；
 - (4)掌握选择结构指令积木的使用；
 - (5)掌握循环结构指令积木的使用；
 - (6)掌握画笔及设置画笔的大小、粗细的操作；
 - (7)掌握控制角色的大小和可视状态的操作；
 - (8)掌握交互相关指令积木的使用；
 - (9)掌握并应用逻辑运算与关系运算的指令积木；
 - (10)能够录入声音，并对声音进行简单处理；
- #### 2.掌握如何应用编程环境中的指令积木实现交互效果的程序
- (1)了解随机数的概念；
 - (2)掌握选择结构、循环结构的流程图画法；
 - (3)掌握程序包含选择结构并处理最多三个条件之间关系的操作；
 - (4)掌握在程序中包含循环结构的操作；
 - (5)掌握在程序中包含交互功能(按下键盘按键、点击角色等)的操作；
 - (6)掌握在程序中能使用画笔绘制图形的操作；
 - (7)掌握在程序中包含侦测场景的操作；

16.2.2 包含的指令积木

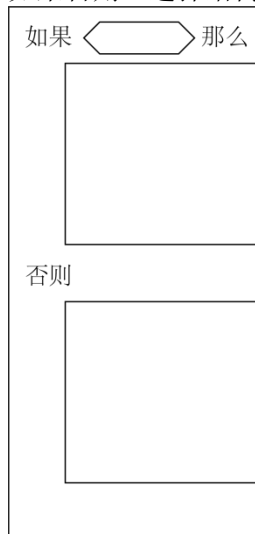
重复执行（循环结构）



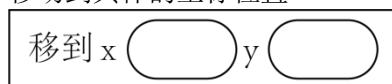
如果（选择结构）



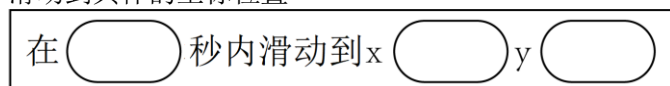
如果否则（选择结构）



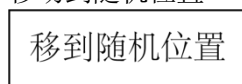
移动到具体的坐标位置



滑动到具体的坐标位置



移动到随机位置



增加角色 x 坐标

x 增加

增加角色 y 坐标

y 增加

增加角色大小

大小增加

角色大小设定为百分之多少

大小设定为 %

显示角色

显示角色

隐藏角色

隐藏角色

角色移到顶层

角色移到顶层

角色移到底层

角色移到底层

角色移动到某一层

角色移动到 层

碰到边缘

碰到边缘

碰到鼠标

碰到鼠标

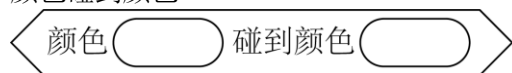
碰到角色

碰到角色

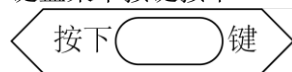
碰到颜色

碰到颜色

颜色碰到颜色



键盘某个按键按下



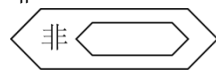
与



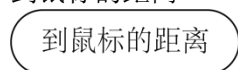
或



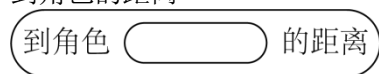
非



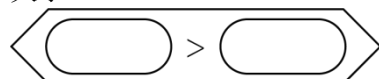
到鼠标的距离



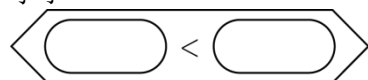
到角色的距离



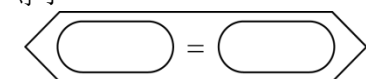
大于



小于



等于



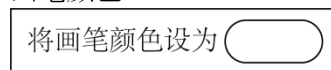
落笔



抬笔



画笔颜色



画笔大小

将画笔大小设为 <input type="text"/> 像素	将画笔大小设为 <input type="text"/> 像素
---------------------------------	---------------------------------

清除画笔内容

清除画笔内容

16.3 三级评价要求

16.3.1 评价目标与能力要求

掌握图形化编程环境的高级功能，应用随机数、变量、逻辑运算与关系运算的组合，解决实际问题；考查对选择语句、循环语句的嵌套使用，以及运用循环简化多次的反复操作程序的理解程度。考查多种情况的逻辑处理和交互控制能力。

具体包括以下2方面能力要求：

1.掌握编程环境的高级功能

- (1)掌握新建、删除变量的操作；
- (2)掌握如何设定变量值以及在舞台区显示、隐藏变量的操作；
- (3)掌握逻辑运算与关系运算的组合使用；
- (4)掌握运用循环简化多次的反复操作的程序。

(5)掌握并应用数学运算的指令积木；

2.理解并在程序中使用随机数和变量

- (1)掌握并能够灵活应用随机数；
- (2)理解变量的概念，理解变量的作用域；
- (3)掌握通过变量的变化让程序跳转到不同的部分的操作；
- (4)掌握在程序中包含不同条件选择语句的嵌套；
- (5)掌握在程序中包含循环语句嵌套的操作；
- (6)掌握如何在程序中包含根据选择语句的真、假跳出循环的程序；
- (7)掌握循环语句、选择语句嵌套的综合运用。

16.3.2 包含的指令积木

随机数

在 <input type="text"/> 到 <input type="text"/> 之间随机取一个数

设定变量的值（在图形化编程环境中创建变量的菜单或按钮）

将变量 <input type="text"/> 的值设为 <input type="text"/>

显示变量

显示变量 <input type="text"/>

隐藏变量

隐藏变量 <input type="text"/>

加

<input type="text"/> + <input type="text"/>

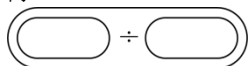
减



乘



除



求余



获取用户输入（会在舞台中弹出一个文本输入框）

获取用户输入

获取系统时间（通过下拉菜单可以选择年、月、日、时、分、秒）

系统时间（年）

16.4 四级评价要求

16.4.1 评价目标与能力要求

考查对图形化编程环境的较强综合操作能力，考查使用图形化编程环境进行数据处理的能力，同时考查对函数和过程的理解和使用能力，以及对已掌握知识的深度综合应用及思考更优程序方案。考查结合分析和计算的情境。

具体包括以下能力要求：

- (1)掌握新建列表的操作并能利用程序实现对列表的数据的插入、删除、查找等操作；
- (2)掌握综合运用字符串指令积木对字符串进行操作；
- (3)理解函数的作用范围；
- (4)掌握创建函数的操作并能正确的添加不同类型的参数；
- (5)运用函数解决实际的问题以及通过函数优化程序；
- (6)了解多线程的概念；
- (7)掌握综合运用复杂的循环实现算法的操作；
- (8)理解排序、递推、递归、分治常见算法的概念并能利用算法解决实际问题。

16.4.2 包含的指令积木

插入列表（在编程环境中创建列表的菜单或按钮）

将 插入列表

删除列表项

删除列表 的第 项

替换列表项

将列表 的第 项替换为

显示列表

显示列表

隐藏列表

隐藏列表

组合字符串

连接 和

(单独) 获取字符串中的 (某一位的) 字符

中的第 个字符

17 评价方法

17.1 评价形式

本指南规定的评价采用上机考试形式。考生需在指定的图形化编程平台中，通过拖拽、组合指令积木的方式，完成理论知识与编程的综合考核。评价内容应侧重于检验考生对编程逻辑的理解、程序结构的搭建以及利用编程解决实际问题的能力。

17.2 评价环境

软件环境：评价机构应提供统一的、稳定的图形化编程平台。该平台需具备指南中规定的核心功能，包括但不限于角色与背景控制、流程控制（顺序、分支、循环）、变量、列表、事件交互及画笔等功能模块。

硬件环境：保证每位考生使用一台性能良好的计算机，并配备鼠标、键盘、耳机等必要外设。考试环境应保证系统运行流畅，画面渲染正常

17.3 题型与内容结构

评价试题由客观题和编程题两部分构成，全面考察考生的理论知识理解和实际编程能力。各等级试题结构建议如下表所示：

等级	客观题（占比）	编程题（占比）	核心考察重点
一级	约60%	约40%	基本操作、顺序结构、角色与背景控制。
二级	约60%	约40%	分支与循环结构、简单交互、事件侦测、逻辑运算。
三级	约40%	约60%	变量使用、循环嵌套、复杂条件判断、数学运算。
四级	约40%	约60%	列表操作、函数（自定义积木）使用、算法思维、模块化编程。

客观题：包括但不限于单项选择题、多项选择题、判断题等。主要用于考察对编程概念、指令功能、程序逻辑流的识记和理解。

编程题：要求考生根据题目要求，在编程平台中设计、完成并调试完整的程序项目，以实现特定功能或解决情境化问题。重点考察程序的功能完整性、逻辑正确性、结构规范性以及一定的创意实现。

17.4 成绩评定

评价结果采用百分制计分。

考生成绩达到规定的合格分数线（通常为 60 分）方视为通过该等级考核。

评价系统或评卷人员应根据预设的、细化的评分标准对编程题进行打分，评分维度应包括功能实现、代码结构、语法正确性等。

17.5 评价监督与核验

评价活动应由具备资质的考官或监考系统进行监督，确保考试的公平、公正。

对于编程题，评价系统应具备编程过程记录或成果文件自动提交功能，以备核验。

对于高级别的编程题，可辅以必要的结果核验或简答面试，以确认考生对复杂代码逻辑的理解。

参 考 文 献

- [1] GB/T 1526-1989 信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定
- [2] GB/T 5271.13 信息技术 词汇 第13部分：计算机图形

团 体 标 准

T/CIE XXX-XXX

青少年软件编程等级评价指南 第 3 部分：C/C++语言编程

Evaluation Guideline for Software Programming Grade of Juniors

Part 3: C/C++ Language Programming

（草案）

2025 – XX – XX 发布

2025 – XX – XX 实施

目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 C/C++编程等级评价指南概述	2
5 C/C++编程等级评价要求	3
6 评价方法	7

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是T/CIE**《青少年软件编程等级评价指南》的第3部分，T/CIE**已发布了以下几个部分。

—第1部分：通则

—第2部分：图形化编程

—第3部分：C/C++语言编程

—第4部分：Python语言编程

本文件按照T/CIE 104.3-2021进行修订。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中国电子学会提出。

本文件由中国电子学会科学普及工作委员会归口。

本文件起草单位：

本文件主要起草人：

青少年软件编程等级评价指南

第3部分：C/C++语言编程

18 范围

本文件规定了青少年C/C++编程的等级、能力要求、评价标准及评价方法。

本文件适用于青少年C/C++编程等级评价所涉及的组织、机构或企业。本文件不适用于以学历教育或课程教学为目的的评价活动。

本部分是T/CIE**《青少年软件编程等级评价指南》的第3部分，主要规定C/C++编程的等级评价内容。该系列标准的其他部分分别规定通则、图形化编程及Python编程的等级评价内容。

19 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 29802-2013 《信息技术 学习、教育和培训 测试试题信息模型》

GB/T 5271.13 数据处理词汇 13部分 计算机图形

GB/T 1526-1989 《信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定》

GB/T 15272 《程序设计语言C》

ISO/IEC 14882-2020 《信息技术.编程语言.C++》

ISO/IEC 9899:2018 《程序设计语言.C 标准》

GB/T 34943-2017 《C/C++语言源代码漏洞测试规范》

T/CIE 098-2019 青少年软件编程等级评价指南 第1部分：通则

20 术语和定义

GB/T 29802、GB/T 5271.13、GB 1526-1989、GB/T 15272、ISO/IEC 14882-2017、ISO/IEC 9899:2018界定的以及下列术语和定义适用于本文件。

20.1

数据结构 data structure

数据组织、管理和存储的方式，能够使数据被高效的访问和修改。

20.2

算法 algorithm

为解决特定问题而规定的一系列有限的操作步骤。

20.3

变量 variable

程序中用于存储数据的、其值可以改变的量，由变量名和数据类型定义。

20.4

函数 function

用于完成特定功能的一段独立的代码单元，可以通过参数接收数据，并可能返回一个结果。

20.5

递归 recursion

一种函数直接或间接调用自身的方法。

20.6

面向对象编程 object-oriented programming; OOP

一种以对象为核心，包含封装、继承和多态三大特性的编程范式。

20.7

时间复杂度 time complexity

定性的描述算法运行时间随输入数据规模增长的变化趋势。

20.8

空间复杂度 space complexity

定性的描述算法在运行过程中临时占用存储空间随输入数据规模增长的变化趋势。

20.9

编译 compile

将用高级编程语言编写的源代码（source code）转换生成目标代码（object code）的过程。

20.10

调试 debug

发现、定位和修正程序代码中错误的过程。

20.11

测试用例 test case

为验证程序特定功能或路径而设计的一组输入数据、执行条件和预期结果。

20.12

在线评测系统 online judge system

一种能够自动编译、运行用户提交的程序，并使用预设测试用例检验其正确性和性能的在线系统。

21 C/C++编程等级评价指南概述

21.1 目的

本文件从软件编程所需要的技能和知识出发，以规范化评价青少年C/C++编程能力，引导地区的组织、机构及企业根据当地编程教育普及情况搭建适合当地的青少年C/C++编程培养模式，从而激发和培养青少年学习编程技术的热情和兴趣。让考生能够掌握C/C++编程的相关语法、算法和数据结构。通过设定不同等级的考试目标，让考生具备一定的程序设计能力，为后期数据挖掘、人工智能等学科的学习打下良好基础。

本文件不以评价教学为目的，考核内容不是按照学校要求设定，而是从软件编程所需要的技能和知识以及国内编程教育的普及情况而定，主要以实践应用能力为主。

21.2 评价对象

本文件适用于6岁-18岁青少年，建议8岁及以上参与。

21.3 等级划分

C/C++编程等级评价体系共设十个等级，从一级至十级，构建了一个从编程启蒙到算法竞赛水平的完整能力发展路径。十个等级划分为四个系统性的进阶阶段，体现了从基础概念掌握到高阶创新应用的递进式能力培养模式：

基础语法奠基阶段（一级-二级）：建立编程基础概念和环境操作能力，掌握核心语法和基本逻辑结构。

入门算法与数据结构阶段（三级-五级）：系统学习基础算法思想和数据结构应用，培养初步的算法思维能力。

进阶算法与数据结构阶段（六级-七级）：深入掌握核心算法和复杂数据结构，提升系统化的问题解决能力。

高阶应用与竞赛阶段（八级-十级）：精通高级算法优化和复杂问题求解，培养算法竞赛水平和创新能力。

每个等级的知识点和能力要求都在前一等级基础上进行扩展和深化，形成有机衔接的完整编程能力体系。

表3 青少年 C/C++编程等级逻辑关系

级别过渡	能力进阶描述
基础语法奠基 → 入门算法与数据结构	从基础语法和流程控制过渡到算法与数据结构基础应用，系统引入函数封装、字符串处理、结构体、排序算法和基础算法思想。
入门算法与数据结构 → 进阶算法与数据结构	从基础算法应用提升到核心算法与数据结构掌握，深入学习搜索算法、动态规划、树形结构和图论基础等核心内容。
进阶算法与数据结构 → 高阶应用与竞赛	从核心算法掌握进阶到高级优化与复杂问题求解，掌握高级数据结构、复杂算法优化技巧和高阶数学理论应用。

21.4 能力要求

表4 各阶段等级分布与能力特征

阶段	包含等级	阶段能力特征	核心培养目标
基础语法奠基阶段	一级-二级	掌握编程环境操作，理解变量、数据类型、流程控制等基础概念，能够编写简单的顺序、分支、循环结构程序。	建立编程基础，培养计算思维
入门算法与数据结构阶段	三级-五级	掌握函数、字符串、结构体等进阶语法，能够应用排序、递归、贪心等基础算法解决实际问题，初步使用STL容器。	培养算法思维，掌握基础数据结构
进阶算法与数据结构阶段	六级-七级	系统掌握搜索算法、动态规划、树结构、图论基础等核心算法与数据结构，能够解决中等复杂度的算法问题。	提升算法设计与实现能力
高阶应用与竞赛阶段	八级-十级	精通高级数据结构（线段树、平衡树等）和复杂算法优化技术，能够运用高阶数学理论解决竞赛级别的复杂问题。	培养创新能力和竞赛水平

22 C/C++编程等级评价要求

22.1 一级评价要求

熟练掌握一种 C/C++ 开发编译环境；初步掌握变量的定义与使用；掌握基本运算语句，输入输出语句的使用方法；熟练掌握顺序结构，分支结构，循环结构的使用。

具体包括以下能力要求：

- a) 理解变量的概念与类型。
- b) 掌握变量的定义，赋值方法。
- c) 掌握基本的输入输出方法。
- d) 掌握基本的运算语句。
- e) 了解常用数学函数并掌握其使用方法。

- f) 能够完成一个顺序结构的程序。
- g) 理解逻辑运算的基本概念
- h) 掌握基本逻辑表达式的功能及写法。
- i) 能够完成一个选择结构的程序。
- i) 掌握基本循环语句的功能及写法。
- k) 能够完成一个循环结构的程序。
- l) 掌握三目运算符的写法。
- m) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.2 二级评价要求

熟练掌握多层分支结构，循环结构嵌套的使用方法；掌握一维数组及二维数组的使用方法；了解素数的概念并初步掌握基础的埃氏筛法。

具体包括以下能力要求：

- a) 掌握变量的类型转换。
- b) 掌握多层分支结构的使用方法。
- c) 能够编写一个多层分支结构的程序。
- d) 掌握多层循环结构的使用方法。
- e) 能够编写一个多层循环结构的程序。
- f) 理解数组的概念与类型。
- g) 掌握一维数组及与二维数组的定义，赋值，查询方法。
- h) 能够完成包含分支结构，循环结构，数组等内容的综合问题。
- i) 理解素数的概念。
- j) 掌握素数的判定方法。
- k) 掌握素数筛的埃氏筛法。
- l) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.3 三级评价要求

掌握字符数组与字符串的使用方法；了解函数的定义；初步掌握函数的调用方法；初步掌握函数参数传递的方法；初步掌握模拟算法和枚举算法；

具体包括以下能力要求：

- a) 掌握字符数组和字符串的定义，赋值方法。
- b) 掌握字符串相关函数的使用方法。
- c) 理解函数的概念。
- d) 掌握函数的定义与调用方法。
- e) 理解函数参数传递的概念。
- f) 掌握函数参数的传递方法。
- g) 理解模拟算法的概念。
- h) 能够使用模拟算法解决较为简单的问题。
- i) 理解枚举算法的概念。
- j) 能够使用枚举算法解决较为简单的问题。
- k) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.4 四级评价要求

掌握结构体的使用方法；掌握基本的进制转换方法；理解位运算的概念并掌握位运算的基本方法；熟练掌握快速幂算法与排序算法；熟练掌握高精度运算；熟练掌握文件读写操作；初步掌握递推算法与递归算法；初步掌握贪心算法；

具体包括以下能力要求：

- a) 理解结构体的概念。
- b) 掌握结构体的使用方法。
- c) 理解进制的概念。

- d) 掌握二进制，八进制，十进制，十六进制间的转换方法。
- e) 了解基本的位运算并掌握基本的位运算语句。
- f) 掌握快速幂算法。
- g) 掌握桶排序，冒泡排序，选择排序，插入排序，快速排序等多种排序方法。
- h) 掌握高精度加法，高精度减法，高精度乘法，高精度除法。
- i) 掌握基本的文件读写操作。
- j) 掌握递推算法与递归算法并能够使用其解决较为简单的问题。
- k) 掌握贪心算法并能够使用其解决较为简单的问题。
- l) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.5 五级评价要求

掌握前缀和算法与差分算法；掌握二分算法与双指针算法；掌握栈与队列的使用；了解 STL；初步掌握 STL 的使用。

具体包括以下能力要求：

- a) 掌握前缀和算法并能够使用其解决较为简单的问题。
- b) 掌握差分算法并能够使用其解决较为简单的问题。
- c) 掌握二分查找与二分答案并能够使用其解决较为简单的问题。
- d) 掌握双指针算法并能够使用其解决较为简单的问题。
- e) 理解栈的概念并掌握其使用方法。
- f) 理解队列的概念并掌握其使用方法。
- g) 了解 STL 中的 vector, set, map。
- h) 掌握 STL 中 vector, set, map 的使用方法。
- i) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.6 六级评价要求

熟练掌握搜索算法；掌握简单动态规划算法；掌握基础数论知识；掌握基础组合数学知识。

具体包括以下能力要求：

- a) 掌握深度优先搜索算法。
- b) 掌握广度优先搜索算法；
- c) 理解动态规划的概念。
- d) 能够使用线性动态规划解决较为简单的问题。
- e) 掌握 01 背包与完全背包算法；
- f) 能够使用背包算法解决较为简单的问题。
- g) 能够使用区间动态规划解决较为简单的问题。
- h) 掌握初中数学的基础知识。
- i) 掌握求最大公约数的欧几里得算法。
- j) 掌握加法原理，乘法原理并能解决简单的问题。
- k) 理解排列与组合的定义。
- l) 掌握排列与组合的计算公式。
- m) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.7 七级评价要求

掌握链表的使用；掌握树的基础知识；熟练掌握较为复杂的动态规划；掌握图论的基础知识；掌握最短路算法；掌握最小生成树算法；掌握字符串哈希算法；掌握倍增算法；

具体包括以下能力要求：

- a) 掌握单向链表，双向链表，循环链表的使用。
- b) 掌握树的定义，存储与遍历。
- c) 掌握二叉树的基础知识。
- d) 了解哈夫曼编码。
- e) 掌握哈夫曼树的基础知识。

- f) 能够使用较为复杂的二维动态规划解决问题。
- g) 掌握动态规划的最值优化并利用其解决问题。
- h) 掌握图的定义，存储与遍历。
- i) 掌握有向无环图的拓扑排序算法；
- j) 掌握最短路的 Floyd 算法，Dijkstra 算法，SPFA 算法。
- k) 掌握最小生成树的 Kruskal 算法，Prim 算法。
- l) 了解哈希表的概念。
- m) 掌握字符串哈希算法。
- n) 掌握倍增算法并能够使用其解决较为简单的问题。
- o) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.8 八级评价要求

掌握 ST 表的使用；掌握部分高级树形数据结构的使用；掌握求欧拉回路的方法；掌握字符串匹配的 KMP 算法；掌握较为复杂的树形动态规划与状态压缩动态规划。

具体包括以下能力要求：

- a) 掌握 ST 表的使用。
- b) 掌握并查集的使用。
- c) 掌握树状数组的使用。
- d) 掌握线段树的使用。
- e) 掌握 trie 树的使用。
- f) 理解欧拉回路的概念。
- g) 掌握求欧拉回路的方法。
- h) 掌握字符串匹配的 KMP 算法。
- i) 掌握树形动态规划的最值优化并利用其解决问题。
- j) 掌握状态压缩动态规划的最值优化并利用其解决问题。
- k) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.9 九级评价要求

熟练掌握图的连通性相关的概念和算法；掌握基础数论算法；掌握矩阵及其基本运算；掌握容斥原理及其应用。

具体包括以下能力要求：

- a) 理解连通分量，强连通图，强连通分量等概念。
- b) 掌握有向图强连通分量的求法。
- c) 掌握无向图割点，割边的求法。
- d) 掌握无向图点双连通分量，边双连通分量的求法。
- e) 掌握缩点算法并利用其解决问题。
- f) 掌握扩展欧几里得算法并能利用其解同余方程。
- g) 理解逆元的概念。
- h) 掌握费马小定理并能利用其求逆元。
- i) 理解矩阵的概念。
- j) 掌握矩阵的加法，减法，乘法原则。
- k) 掌握矩阵快速幂算法并能利用其解决问题。
- l) 掌握容斥原理并能利用其解决问题。
- m) 能够使用上述方法编写完成指定功能的正确完整的程序。

22.10 十级评价要求

掌握至少一种平衡树的使用；掌握较为复杂的动态规划优化方法；掌握部分较为复杂的数论知识。

具体包括以下能力要求：

- a) 掌握至少一种平衡树的使用，包括但不限于 treap, splay 等。
- b) 掌握复杂动态规划的状态设计方法并能利用其解决问题。

- c) 掌握动态规划的斜率优化方法。
- d) 掌握动态规划的决策单调性优化方法。
- e) 掌握欧拉定理并能利用其解决问题。
- f) 掌握扩展欧拉定理并能利用其解决问题。
- g) 掌握中国剩余定理并能利用其解同余方程组。
- h) 能够使用上述方法编写完成指定功能的正确完整的程序。

23 评价方法

23.1 评价形式

本指南规定的评价采用上机考试形式。考生需在规定的集成开发环境（IDE）或代码编辑器中，完成理论知识与编程实践的综合性考核。重点考察考生的计算思维、对 C/C++ 语法、数据结构与算法的理解、应用以及代码实现能力。

23.2 评价环境

软件环境：评价机构应提供统一的、稳定的 C/C++ 编程环境。

硬件环境：保证每位考生使用一台性能良好的计算机，确保程序编译与运行流畅。考试环境需具备稳定的电力与网络（如需在线提交或使用在线评测系统）。

23.3 题型与内容结构

评价试题由客观题和编程实操题两部分构成，各阶段的试题结构和考察重点应有所区别：

阶段	等级范围	客观题 (占比)	编程题 (占比)	编程题 数量	考察重点
基础语法奠基阶段	一级-二级	60%	40%	2题	基础语法、程序结构（顺序、分支、循环）、简单数组应用
入门算法与数据结构阶段	三级-五级	40%	60%	2题	函数、标准模板库(STL)、基础算法（排序、枚举、递归）
进阶算法与数据结构阶段	六级-七级	20%	80%	4题	数据结构（树、图）、复杂算法（动态规划、搜索、图论）。
高阶应用与竞赛阶段	八级-十级	0%	100%	4题	高级数据结构与算法（线段树、数论、动态规划优化）

客观题：包括但不限于单项选择题、多项选择题、判断题等。主要用于考察学生的计算思维能力，以及对语法规则、数据结构概念、算法原理的识记和理解。

编程实操题：要求考生根据题目要求，编写、调试并通过一个或多个完整的 C/C++ 程序。重点考察代码的正确性（能否通过测试用例）、效率（时间与空间复杂度）、规范性（代码结构、命名、注释）以及逻辑严谨性。高等级题目通常涉及具体的算法实现与问题建模。

23.4 成绩评定

评价结果采用百分制计分。

考生成绩达到规定的合格分数线（原则上为 60 分）方视为通过该等级考核。

编程题的评分应优先采用自动化评测系统（Online Judge, OJ），根据预设的测试用例对程序的正确性和性能进行客观评分。对于无法完全自动化评分的部分（如代码规范、思路创新），应由评卷人员根据细化标准进行人工复核。

23.5 评价监督与核验

评价活动应由具备资质的考官或监考系统进行全程监督，确保考试的公平、公正。

对于编程题，强烈建议使用在线评测系统（OJ）进行考核与核验，该系统能自动记录提交历史、编译错误、运行结果和得分情况，形成可追溯的电子档案。

对于八级及以上的高级别考核，因其算法复杂度高，可辅以简单的面试或代码讲解环节，以核验考生对复杂代码逻辑的真实理解程度和问题分析能力。

团 体 标 准

T/CIE XXX-XXX

青少年软件编程等级评价指南 第 4 部分：Python 语言编程

Evaluation Guideline for Software Programming Grade of Juniors

Part 4: Python language Programming

（草案）

2025 – XX – XX 发布

2025 – XX – XX 实施

目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 Python 编程等级评价指南概述	1
5 Python 编程等级评价标准	2
6 评价方法	5
参考文献	6

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是T/CIE**《青少年软件编程等级评价指南》的第4部分，T/CIE**已发布了以下几个部分。

—第1部分：通则

—第2部分：图形化编程

—第3部分：C/C++语言编程

—第4部分：Python语言编程

本文件由中国电子学会提出。

本文件由中国电子学会科学普及工作委员会归口。

本文件起草单位：

本文件主要起草人：

青少年软件编程等级评价指南

第4部分：Python 语言编程

24 范围

本文件规定了青少年Python编程的等级、能力要求、评价标准及评价方法。

本文件适用于青少年Python编程等级评价所涉及的组织、机构或企业。本文件不适用于以学历教育或课程教学为目的的评价活动。

25 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 29802 信息技术 学习、教育和培训测试试题信息模型

GB/T 5271.13 信息技术 词汇 第13部分：计算机图形

GB 1526-1989 信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定

T/CIE 098-2019青少年软件编程等级评价指南 第1部分：通则

26 术语和定义

GB/T 29802、GB/T 5271.13、GB 1526-1989界定的下列术语和定义适用于本文件。

26.1

语句式脚本

由 Python 基础语句构成的程序代码，以不包含函数、类、模块等语法元素为特征。

26.2

模块式脚本

由 Python 语句、函数、类、模块等元素构成的程序代码，以支持函数、类、模块的创建及使用为特征。

26.3

IDLE

由 Python 官方网站（<https://www.python.org>）所提供 Python 软件自带的集成开发环境。

27 Python 编程等级评价指南概述

27.1 评价目的

本文件从软件编程所需要的技能和知识出发，以规范化评价青少年Python编程为中心，引导地区的组织、机构及企业根据当地编程教育普及情况，搭建适合当地的青少年Python编程培养模式，从而激发和培养青少年学习编程技术的热情和兴趣。让考生能够掌握Python编程的相关知识和操作能力，熟悉编程各项基础知识和理论框架，通过设定不同等级的考试目标，让考生具备Python编程能力，为后期大数据处理与人工智能编程等专业化编程学习打下良好基础。

本文件不以评价教学为目的，考核内容不是按照学校要求设定，而是根据Python编程所需能力以及国内Python编程教育的普及情况而定，主要以实践应用能力为主。

27.2 评价对象

本文件适用于6岁-18岁青少年，建议8岁及以上参与。

27.3 等级划分

青少年Python编程共分为六个等级，从一级到六级，依次从编程基础入门到具备初步的数据结构与算法能力。级别之间的逻辑关系及等级对应的能力描述如表1和表2所示，文件第五章中描述了每个等级相应的核心知识点和对知识点的掌握程度要求。

六个等级之间呈现逐级递进、能力叠加的关系：

低等级（一、二级）注重基础语法和流程控制，为后续学习打下基础。

中等级（三、四级）引入算法思维和模块化编程，提升代码组织和问题分解能力。

高等级（五、六级）强调数据处理、系统设计和数据结构，为专业化编程学习做准备。

每个等级的知识点和能力要求都是在前一等级基础上扩展和深化，形成完整的编程能力体系。

表5 青少年 Python 软件编程等级逻辑关系

级别	逻辑点
一级→二级	从基础语法和顺序结构过渡到核心数据类型和循环/分支结构，增强程序控制能力。
二级→三级	从数据类型操作提升到算法实现，引入推导式和内置函数，培养算法思维。
三级→四级	从算法基础过渡到函数封装和模块化设计，引入递归、分治和标准库使用。
四级→五级	从模块化编程扩展到文件操作、数据处理和可视化，引入数据库基础。
五级→六级	从数据处理提升到面向对象编程和数据结构，掌握类、线性与非线性结构及其算法。

27.4 能力要求

表6 青少年 Python 软件编程等级能力要求

等级	能力描述	核心知识点
一级	熟悉编程环境，具备编写顺序与单分支结构的简单程序的基本编程能力	简单数学运算与Turtle库
二级	掌握核心数据类型的概念与操作，具备编写顺序、分支、循环结构的简单程序的基本编程能力	核心数据类型：列表、元组、字符串、字典、集合
三级	具有基本的算法思维，具备以算法为目标的基本编程能力	推导式，常用内置函数，算法：解析、枚举、排序，数制转换
四级	具有初步的模块编程思维，具备以函数形式代码复用的基本编程能力	自定义函数，递归与分治，查找，模块及常用标准库
五级	具有初步的数据意识与思维，具备以数据处理为目的的基本编程能力	简单文件读写、数据统计与可视化、数据库
六级	掌握简单的数据结构概念与操作，具备选择合理的数据结构与算法编程处理问题的基本编程能力	类与对象，简单数据结构与算法：数组、字符串、队列、栈、二叉树

28 Python 编程等级评价标准

28.1 一级评价要求

掌握基本的 Python 编程相关知识和方法，会使用 IDLE 进行编程，熟悉 Python 语言的基本语法规则，会用 turtle 库完成简单的顺序执行的 Python 程序，能够解决较为简单的问题。参加本级考核的考生，对 Python 编程有基本的了解，熟悉至少一种 Python 编程环境的操作，会编写含有变量及库文件的基本程序。具备编写顺序结构的简单程序的基本编程能力。

具体包括以下能力要求：

1. 了解 Python 多种开发环境，具备使用 Python 开发环境进行简单程序编写的能力
 - (1) 了解 Python 常见的几种编程环境：IDLE、thierry、Visual Studio Code、JupyterNotebook；
 - (2) 了解 Python 的两个版本号以及迭代方向；
 - (3) 掌握 IDLE 的操作过程，会打开 IDLE，会新建文件、保存文件；
 - (4) 掌握使用 IDLE 进行编程，会修改文件、运行文件等操作；
 - (5) 掌握 IDLE 的两种开发模式，会在不同模式下进行切换。

2.掌握 Python 程序编写的基本方法

- (1) 理解“输入、处理、输出”程序编写方法；
- (2) 掌握 Python 编程的基本格式，编写脚本程序时会使用缩进、注释、字符串标识等；
- (3) 掌握变量的基本概念，掌握变量名的命名和保留字等基本语法，理解变量赋值；
- (4) 理解字符串、数值型变量，会对变量类型进行转换；
- (5) 掌握并熟练编写带有数值类型变量的程序，具备解决数学运算基本问题的能力；
- (6) 理解算术、关系、逻辑运算符、表达式及运算的基本概念，掌握 Python 编程基础的逻辑运算，数学运算中不涉及开方运算。

3.具备基本的计算思维能力，能够完成简单的 Python 程序编写

- (1) 理解顺序、单分支结构语句的特点，能够完成简单的顺序与单分支结构程序编写；
- (2) 理解运算符、表达式、逻辑运算的基本概念，掌握 Python 编程基础的逻辑表达式；
- (3) 了解第三方库 turtle 的功能，会导入该库文件，掌握它的一些简单使用方法：前进、后退、左右转、提落笔、画点、画圆等。

28.2 二级评价要求

掌握 Python 语言的常用核心数据类型以及编程流程控制；会使用分支和循环结构语句完成简单的 Python 程序编写，能够编程解决相应的问题。参加本级考核的考生，对 Python 编程有较深入的了解，熟悉 Python 语言的常用核心数据类型以及编程流程控制语句。具备编写顺序、分支、循环结构的简单程序的基本编程能力。

具体包括以下能力要求：

1.知道与掌握 Python 编程的进阶知识

- (1) 知道如下数据类型：列表、元组、字符串、字典、集合五大内置核心数据类型；
- (2) 理解列表类型的概念，掌握它的基础用法及操作：声明、访问、获取元素个数、遍历、添加和删除、连接、排序等；
- (3) 理解元组类型的概念，掌握它的基础用法及操作：声明、访问、获取元素个数、遍历等；
- (4) 理解字典类型的概念，掌握它的基础用法及操作：声明、访问、获取元素个数、追加元素、删除元素等；
- (5) 理解字符串类型的概念，掌握它的基础用法及操作：声明、访问、获取长度、遍历、分割、组合、替换、格式化等；
- (6) 理解集合类型的概念，掌握它的基础用法及操作：声明、获取长度、交集、并集、追加元素、更新。

2.会编写较为复杂的 Python 程序并掌握 Python 编程的控制语句

- (1) 理解选择结构语句的功能和写法，能够完成简单选择结构的程序编写；
- (2) 掌握程序的单分支结构，理解二分支、多分支结构语句；
- (3) 理解循环结构语句的功能和写法，能够完成简单循环结构的程序编写；
- (4) 理解 for 循环、while 循环、break 和 continue 循环控制结构语句。

28.3 三级评价要求

理解 Python 语言常用算法的基本框架，能够用 Python 语言实现常用算法的编程。能够使用和处理相关数据，解决较为复杂的编程问题。参加本级考核的考生，对 Python 编程有初步的系统理解，初步掌握 Python 语言的算法编程。具有基本程序的 Debug 能力，具备以算法为目标的基本编程能力。

具体包括以下能力要求：

1.掌握各类推导式的使用方法，具备快速处理可迭代对象、编写更优雅代码的能力

- (1) 掌握列表推导式、字典推导式、生成器推导式的使用方法；
- (2) 掌握序列解包和切片的使用方法。

2.掌握常用核心内置函数的功能及用法。

3.具备基础的算法编程能力

- (1) 理解算法的概念，掌握解析、枚举、排序算法的特征及其程序实现；
- (2) 理解编码、数制的基本概念，能够解决相关问题。能够进行二进制、八进制、十六进制与十

进制之间的相互转换，理解 Python 中的数制转换函数。

28.4 四级评价要求

认识函数的概念，熟悉函数的相关操作，掌握自定义函数的创建与调用；理解递归与递推、分治（对分查找）算法的思想，能够用递归与递推、分治算法编程解决生活问题；理解算法的优化方法；掌握模块及常用标准库的功能、获取、安装与调用方法。参加本级考核的考生，能够利用函数与自定义函数优化程序结构，能够用递归与递推、分治算法编写程序与软件，能够调用 Python 的模块及常用标准库解决问题。具有初步的模块编程思维，具备以函数形式代码复用的基本编程能力。

具体包括以下能力要求：

1.掌握自定义函数的应用

- （1）理解函数、函数的参数、函数的返回值、变量的作用域等概念；
- （2）掌握简单自定义函数的创建及调用；
- （3）理解基本算法中递归的概念，实现基本算法中的递归方法；
- （4）理解基本算法中的分治（对分查找）算法，能够用分治算法实现简单的 Python 程序。

2.掌握模块及常用标准库的功能与用法，具备灵活运用已有方法快速扩展程序功能的能力

- （1）掌握 math、random、time 等常用标准库的功能与使用方法；
- （2）理解 jieba、wordcloud 等第三方库的获取与使用方法；
- （3）能够使用基本数据类型和常用模块（库）编写指定功能的程序。

28.5 五级评价要求

掌握 Python 语言中文件的读写操作，利用 pandas 模块进行初步的数据处理，掌握基本数据可视化操作方法。初步掌握 SQLite 数据库基础编程方法。参加本级考核的考生，具备使用 Python 语言进行初步数据处理的能力，掌握简单数据库编程的能力，利用类与对象、数据可视化等知识进一步提高对软件编程的综合能力。

具体包括以下能力要求：

1.掌握 Python 文件读写操作，具备初步的数据处理能力

- （1）理解文件的编码、文本文件和二进制文件，掌握文件的读取、写入、追加与定位操作；
- （2）掌握一维、二维数据的表示、存储和处理方法。

2.掌握基本的数据可视化操作方法，具备初步的数据可视化处理能力

- （1）掌握 pandas 模块的简单使用方法；
- （2）掌握 matplotlib 库的简单使用方法。

3.掌握 SQLite 数据库基础编程方法，初步掌握数据库管理数据的编程能力

- （1）掌握 SQLite 数据库的创建与简单查询、数据库的连接与关闭、创建游标等操作；
- （2）掌握游标对象的方法：execute()、fetchone()、fetchmany()、fetchall()和 close()；
- （3）能够使用上述基本方法编写指定功能的正确完整的程序。

28.6 六级评价要求

掌握 Python 类与对象的概念与使用。掌握 Python 的简单数据结构，了解线性数据结构的概念与特性，掌握其基本操作方法。掌握非线性结构树与二叉树的概念及性质，能够遍历二叉树。能结合算法和数据结构完成编程，解决实际问题。参加本级考核的学生，能够通过创建类与子类编程解决问题；掌握 Python 数据结构：数组、字符串、队列和栈，理解这些数据结构的概念和特性，掌握它们的操作方法及其程序实现，掌握二叉树的特性和遍历方法。能够利用这些数据结构编程解决实际问题。

具体包括以下能力要求：

1.理解类与对象的概念，初步掌握类与对象的使用方法，初步了解面向对象编程

- （1）理解类与实例、属性与方法等面向对象编程的概念；
- （2）理解创建类、创建子类、创建类实例的方法。

2.掌握 Python 线性数据结构，能够初步利用数据结构编程解决问题

- （1）了解数组的概念与特性、掌握数组的基本操作（创建、访问、插入与删除等）和常用操作（合并等）；
- （2）了解字符串的概念与特性、掌握字符串的基本操作（创建、访问、插入与删除等）及其综合

应用；

- (3) 了解队列的概念与特性、掌握队列的基本操作（创建、访问、插入与删除等）及其实现方法；
- (4) 了解栈的概念与特性、掌握栈的基本操作（创建、访问、插入与删除等）及其实现方法；
- 3. 了解非线性数据结构树概念，掌握二叉树的概念与性质，掌握二叉树的遍历方法。

29 评价方法

29.1 评价形式

本指南规定的评价采用上机考试形式，考生需在规定的集成开发环境（IDE）中完成理论知识和编程的综合考核。

29.2 评价环境

软件环境：评价机构提供统一的、稳定的 Python 编程环境。

硬件环境：保证每位考生使用一台性能良好的计算机，并确保网络稳定。

29.3 题型与内容结构

评价试题由客观题和编程题两部分构成，全面考察考生的理论知识理解和实际编程能力。各等级试题结构建议如下表所示：

等级	客观题（占比）	编程题（占比）	核心考察重点
一级至二级	约60%-70%	约40%-30%	基础语法、流程控制、核心数据类型的基本操作。
三级至四级	约60%-70%	约40%-30%	算法思维、函数使用、模块应用、代码调试能力。
五级至六级	约50%-70%	约50%-30%	数据处理、系统设计、数据结构与算法综合应用能力。

客观题：包括但不限于单项选择题、多项选择题、判断题等，主要用于考察对基本概念、语法、原理的识记和理解。

编程实操题：要求考生根据题目要求，编写、调试并运行完整的 Python 程序或代码片段，以解决特定问题。重点考察代码的正确性、规范性、逻辑性和效率。

29.4 成绩评定

评价结果采用百分制计分。

考生成绩达到规定的合格分数线（通常为 60 分）方视为通过该等级考核。

评价系统或评卷人员应根据预设的、细化的评分标准对编程题进行打分，评分维度应包括功能实现、代码结构、语法正确性等。

29.5 评价监督与核验

评价活动应由具备资质的考官或监考系统进行监督，确保考试的公平、公正。

对于高级别的编程题，可辅以必要的结果核验或简答面试，以确认考生对复杂代码逻辑的理解。

参 考 文 献

- [3] [美]David M.Beazley著 谢俊等译《Python参考手册（第4版）》 人民邮电出版社
- [4] 《普通高中信息技术课程标准（2017年版2020年修订）》