

# CRADLE: Empowering Foundation Agents Towards General Computer Control

## CRADLE: 赋能基础代理实现通用计算机控制

Weihao Tan<sup>3</sup>; Wentao Zhang<sup>3</sup>; Xinrun Xu<sup>5</sup>; Haochong Xia<sup>3</sup>; Ziluo Ding<sup>2</sup>; Boyu Li<sup>2</sup>; Bohan Zhou<sup>4</sup>; Junpeng Yue<sup>4</sup>; Jiechuan Jiang<sup>4</sup>; Yewen Li<sup>3</sup>; Ruyi An<sup>3</sup>; Molei Qin<sup>3</sup>; Chuqiao Zong<sup>3</sup>; Longtao Zheng<sup>3</sup>; Yujie Wu<sup>1</sup>; Xiaoqiang Chai<sup>1</sup>; Yifei Bi<sup>2</sup>, Tianbao Xie<sup>6</sup>, Pengjie Gu<sup>3</sup>, Xiyun Li<sup>2</sup>, Ceyao Zhang<sup>7</sup>, Long Tian<sup>1</sup>, Chaojie Wang<sup>1</sup>, Xinrun Wang<sup>3</sup>; Börje F. Karlsson<sup>2</sup>; Bo An<sup>3,18</sup>, Shuicheng Yan<sup>1§</sup>, Zongqing Lu<sup>4,2§</sup>; 谭伟豪<sup>3</sup>; 张文涛<sup>3</sup>; 徐欣润<sup>5</sup>; 夏浩冲<sup>3</sup>; 丁子洛<sup>2</sup>; 李博宇<sup>2</sup>; 周博涵<sup>4</sup>; 岳俊鹏<sup>4</sup>; 姜杰川<sup>4</sup>; 李业文<sup>3</sup>; 安如意<sup>3</sup>; 秦墨磊<sup>3</sup>; 宗楚乔<sup>3</sup>; 郑龙涛<sup>3</sup>; 吴宇杰<sup>1</sup>; 柴晓强<sup>1</sup>; 一飞 Bi<sup>2</sup>, 天宝 Xie<sup>6</sup>, 鹏杰 Gu<sup>3</sup>, 熙云 Li<sup>2</sup>, 张策尧<sup>7</sup>, 田龙<sup>1</sup>, 王超杰<sup>1</sup>, 王欣润<sup>3</sup>; 博耶·F·卡尔松 (Börje F. Karlsson)<sup>2</sup>; 安博<sup>3,18</sup>, 闫水成<sup>1§</sup>, 卢宗庆<sup>4,2§</sup>

<sup>1</sup> Skywork AI <sup>2</sup> Beijing Academy of Artificial Intelligence

<sup>1</sup> Skywork AI <sup>2</sup> 北京人工智能研究院

<sup>3</sup> Nanyang Technological University, Singapore <sup>4</sup> Peking University

<sup>3</sup> 新加坡南洋理工大学 <sup>4</sup> 北京大学

<sup>5</sup> Institute of Software, Chinese Academy of Sciences

<sup>5</sup> 中国科学院软件研究所

<sup>6</sup> The University of Hong Kong <sup>7</sup> The Chinese University of Hong Kong, Shenzhen [weihao001@ntu.edu.sg](mailto:weihao001@ntu.edu.sg) [boan@ntu.edu.sg](mailto:boan@ntu.edu.sg) [zongqing.lu@pku.edu.cn](mailto:zongqing.lu@pku.edu.cn) Project website: <https://baai-agents.github.io/Cradle/>

<sup>6</sup> 香港大学 <sup>7</sup> 香港中文大学（深圳） [weihao001@ntu.edu.sg](mailto:weihao001@ntu.edu.sg) [boan@ntu.edu.sg](mailto:boan@ntu.edu.sg) [zongqing.lu@pku.edu.cn](mailto:zongqing.lu@pku.edu.cn) 项目网站: <https://baai-agents.github.io/Cradle/>



Figure 1: The CRADLE framework empowers nascent foundation models to perform complex computer tasks via the same unified interface humans use, i.e., screenshots as input and keyboard & mouse operations as output.  
图1: CRADLE框架使新兴基础模型能够通过人类使用的统一接口执行复杂的计算机任务，即以截图作为输入，以键盘和鼠标操作作为输出。

## 1 Abstract

## 2 摘要

Despite the success in specific scenarios, existing foundation agents still struggle to generalize across various virtual scenarios, mainly due to the dramatically different encapsulations of environments with manually designed observation and action spaces. To handle this issue, we propose the General Computer Control (GCC) setting to restrict foundation agents to interact with software through the most unified and standardized interface, i.e., using screenshots as input and keyboard and mouse actions as output. We introduce CRADLE, a modular and flexible LMM-powered framework, as a preliminary attempt towards GCC. Enhanced by six key modules: Information Gathering, Self-Reflection, Task Inference, Skill Curation, Action Planning, and Memory, CRADLE is able to understand input screenshots and output executable code for low-level keyboard and mouse control after high-level planning, so that CRADLE can interact with any software and complete long-horizon complex tasks without relying on any built-in APIs. Experimental results show that CRADLE exhibits remarkable generalizability and impressive performance across four previously unexplored commercial video games, five software applications, and a comprehensive benchmark, OSWorld. To our best

尽管在特定场景中取得了成功，现有基础代理仍难以在各种虚拟场景中实现泛化，主要原因在于环境的封装差异巨大，且观察和动作空间多由人工设计。为解决此问题，我们提出了通用计算机控制（General Computer Control，GCC）设置，限制基础代理通过最统一和标准化的接口与软件交互，即使用截图作为输入，键盘和鼠标动作作为输出。我们引入了CRADLE，一种模块化且灵活的基于大语言模型（LMM）的框架，作为迈向GCC的初步尝试。CRADLE由六个关键模块增强：信息收集、自我反思、任务推断、技能策划、动作规划和记忆，能够理解输入截图并在高层规划后输出可执行的低层键盘和鼠标控制代码，从而无需依赖任何内置API即可与任何软件交互并完成长时复杂任务。实验结果表明，CRADLE在四款此前未探索的商业视频游戏、五款软件应用及综合基准OSWorld上展现出卓越的泛化能力和优异性能。我们尽最大努力

---

\*Equal contribution † Core contribution ‡ Equal advising § Corresponding authors

\*同等贡献 † 核心贡献 ‡ 同等指导 § 通讯作者

Weihao Tan's work was conducted during his internships at Skywork AI and BAAI. Longtao Zheng is also an intern at Skywork AI. Xinrun Xu, Bohan Zhou, and Junpeng Yue are interns at BAAI.

谭伟豪的工作在Skywork AI和BAAI实习期间完成。郑龙涛也是Skywork AI的实习生。徐欣润、周博涵和岳俊鹏是BAAI的实习生。

---

knowledge, CRADLE is the first to enable foundation agents to follow the main storyline and complete 40-minute-long real missions in the complex AAA game Red Dead Redemption 2 (RDR2). CRADLE can also create a city of a thousand people in Cities: Skylines, farm and harvest parsnips in Stardew Valley, and trade and bargain with a maximal weekly total profit of 87% in Dealer's Life 2. CRADLE can not only operate daily software, like Chrome, Outlook, and Feishu, but also edit images and videos using Meitu and CapCut. With a unified interface to interact with any software, CRADLE greatly extends the reach of foundation agents by enabling the easy conversion of any

software, especially complex games, into benchmarks to evaluate agents' various abilities and facilitate further data collection, thus paving the way for generalist agents.

知识方面，CRADLE是首个使基础智能体能够遵循主线剧情并完成复杂AAA游戏《荒野大镖客救赎2》（Red Dead Redemption 2, RDR2）中长达40分钟真实任务的系统。CRADLE还可以在《城市：天际线》（Cities: Skylines）中创建千人城市，在《星露谷物语》（Stardew Valley）中种植和收获欧防风，并在《Dealer's Life 2》中进行交易和讨价还价，最高周利润达87%。CRADLE不仅能操作日常软件，如Chrome、Outlook和飞书，还能使用美图秀秀（Meitu）和剪映（CapCut）编辑图片和视频。通过统一接口与任何软件交互，CRADLE极大地扩展了基础智能体的应用范围，使任何软件，尤其是复杂游戏，能够轻松转化为评估智能体多种能力的基准，并促进进一步数据收集，从而为通用智能体铺平道路。

### 3 1 Introduction

#### 4 1 引言

Artificial General Intelligence (AGI) has long been a north-star goal for the AI community [39]. The recent success of foundation agents, i.e., agents empowered by large multimodal models (LMMs) and advanced tools, in various environments, e.g., web browsing [13, 20, 22, 74-76], operating mobile applications [58, 69] and desktop software [63, 73], crafting and exploration in Minecraft [57, 60, 61], and some robotics scenarios [6, 7, 14, 24], have shown promise. However, current foundation agents still struggle to generalize across different scenarios, primarily due to the dramatic differences in the encapsulation of environments with human-designed observation and action space. Therefore, developing foundation agents applicable to various environments remains extremely challenging.

人工通用智能（Artificial General Intelligence, AGI）长期以来一直是人工智能领域的北极星目标[39]。近期，基础智能体，即由大型多模态模型（Large Multimodal Models, LMMs）和先进工具赋能的智能体，在多种环境中取得了成功，例如网页浏览[13, 20, 22, 74-76]、移动应用操作[58, 69]和桌面软件使用[63, 73]、《我的世界》（Minecraft）中的制作与探索[57, 60, 61]，以及部分机器人场景[6, 7, 14, 24]，展现出良好前景。然而，目前的基础智能体仍难以跨场景泛化，主要原因在于环境封装中人类设计的观测和动作空间存在巨大差异。因此，开发适用于多种环境的基础智能体仍极具挑战性。

Computers, as the most important and universal interface that connects humans and the increasingly digital world, provide countless rich software, including applications and realistic video games for agents to interact with, while avoiding the challenges of robots in reality, such as hardware requirements, constraints of practicability, and possible catastrophic failures [48]. Mastering these virtual environments is a promising path for foundation agents to achieve generalizability. Therefore, we propose the General Computer Control (GCC) setting:

计算机作为连接人类与日益数字化世界的最重要且通用的接口，提供了无数丰富的软件，包括应用程序和逼真的电子游戏，供智能体交互，同时避免了现实中机器人面临的硬件需求、实用性限制及潜在灾难性故障等挑战[48]。掌握这些虚拟环境是基础智能体实现泛化能力的有希望路径。因此，我们提出通用计算机控制（General Computer Control, GCC）设定：

Building foundation agents that can master ANY computer task via the universal human-style interface by receiving input from screens and audio and outputting keyboard and mouse actions.

构建能够通过统一的人类风格接口，接收屏幕和音频输入并输出键盘和鼠标操作，掌握任何计算机任务的基础智能体。

There are many challenges to achieving GCC: i) good alignment across multi-modalities for better understanding and decision-making; ii) precise control of keyboard and mouse to interact with the computer, which has a large, hybrid action space, including not only which key to press and where the mouse to move, but also the duration of the press and the speed of the mouse movement; iii) long-horizontal reasoning due to the partial observability of complex GCC tasks, which also leads to the demand for long-term memory to maintain past useful experiences; and iv) efficient exploration in a structured manner to discover better strategies and solutions autonomously, i.e., self-improving, which can allow agents to generalize across the myriad tasks in the digital world.

实现GCC面临诸多挑战：i) 多模态间良好对齐以提升理解和决策能力；ii) 精确控制键盘和鼠标以与计算机交互，

动作空间庞大且混合，不仅包括按键选择和鼠标移动位置，还涉及按压时长和鼠标移动速度；iii) 由于复杂GCC任务的部分可观测性，需进行长时推理，同时要求具备长期记忆以维持过去有用经验；iv) 以结构化方式高效探索以自主发现更优策略和解决方案，即自我提升，从而使智能体能够泛化至数字世界中的众多任务。

As shown in Figure 1, we introduce CRADLE, a novel modular LMM-powered framework that empowers foundation agents towards GCC. CRADLE consists of six key modules: 1) information gathering, to extract the relevant information from multimodal observations; 2) self-reflection, to rethink past experiences about whether the actions and tasks are successfully completed and reasons for possible failures; 3) task inference, to determine whether to continue current tasks or propose a new task given the current situation; 4) skill curation, for generating, updating, and retrieving useful skills for the current task; 5) action planning, to generate specific executable operations for keyboard and mouse control via skills; and 6) memory, for storage, summary, and retrieval of past experiences.

如图1所示，我们提出CRADLE，一种新颖的模块化大型多模态模型驱动框架，赋能基础智能体实现GCC。

CRADLE包含六个关键模块：1) 信息收集，从多模态观测中提取相关信息；2) 自我反思，重新审视过去经验，评估动作和任务是否成功完成及失败原因；3) 任务推断，根据当前情况决定继续当前任务或提出新任务；4) 技能策划，生成、更新和检索当前任务所需的有用技能；5) 动作规划，通过技能生成具体可执行的键盘和鼠标控制操作；6) 记忆，用于存储、总结和检索过去经验。

As illustrated in Figure 2, tasks in GCC can be broadly divided into two categories: video game playing and software application manipulation. video games offer the most challenging tasks in GCC due to several key factors. First, the complexity of game environments requires sophisticated problem-solving and adaptive strategies. Second, long-term reasoning is essential to navigate and succeed in these intricate virtual worlds. Third, understanding and mastering new, complex mechanics within games demand rapid learning and cognitive flexibility. Finally, video games test a player's ability to react quickly and perform precise control and operations, which together create a unique and demanding computational challenge. In addition to the typical embodied control, classical UI manipulation, like menu use, is also common during gameplay, which is similar to the other software applications [48]. Therefore, games provide rich comprehensive and challenging testbeds to

如图2所示，GCC中的任务大致可分为两类：电子游戏玩法和软件应用操作。电子游戏因若干关键因素成为GCC中最具挑战性的任务。首先，游戏环境的复杂性要求复杂的问题解决和适应性策略。其次，长时推理对于在这些复杂虚拟世界中导航和取得成功至关重要。第三，理解并掌握游戏中新颖复杂的机制需要快速学习和认知灵活性。最后，电子游戏考验玩家快速反应和精确控制操作的能力，这些共同构成了独特且严苛的计算挑战。除典型的具身控制外，游戏过程中也常见经典的用户界面操作，如菜单使用，这与其他软件应用类似[48]。因此，游戏为



Figure 2: Taxonomy of GCC and the games and software investigated in this work.

图2: GCC的分类及本研究中考察的游戏和软件。

evaluate and improve agents' various abilities. In this work, we conduct extensive experiments to demonstrate the generalizability of CRADLE in such complex environments, while also mastering diverse everyday software applications in distinct domains. We managed to prove that commercial software is out-of-box testbeds under our framework. The four selected representative games are:

评估和提升智能体的多种能力。在本研究中，我们进行了大量实验，以展示CRADLE在复杂环境中的泛化能力，同时掌握不同领域的多样化日常软件应用。我们成功证明了商业软件在我们的框架下即开即用的测试平台。所选的四款代表性游戏为：

- Red Dead Redemption 2 (RDR2), an epic AAA 3D role-playing game (RPG) with rich storylines, realistic scenes, and an immersive open-ended world; where players can complete missions by following the instructions, freely explore the world, interact with non-player characters (NPCs) and engage in a variety of activities such as hunting and fishing, in a first- or third-person perspective. This game offers great challenges in 3D embodied navigation and interaction.  
• 《荒野大镖客救赎2》 (Red Dead Redemption 2, 简称RDR2)，一款史诗级AAA 3D角色扮演游戏 (RPG)，拥有丰富的剧情、逼真的场景和沉浸式的开放世界；玩家可通过完成任务指令，自由探索世界，与非玩家角色 (NPC) 互动，并以第一人称或第三人称视角参与狩猎、钓鱼等多种活动。该游戏在3D具身导航和交互方面具有极大挑战性。
- Stardew Valley, a 2D pixel-art farming simulation game where players can restore and expand a farm through carefully planned activities such as planting crops, mining, fishing, and crafting. Players can build relationships with the villagers, participate in seasonal events, and uncover the mysteries of the valley. The game encourages strategic planning and time management, as each day brings new opportunities and challenges. Players have to balance their energy and resources to maximize their farm's productivity and profitability.  
• 《星露谷物语》 (Stardew Valley)，一款2D像素风格的农场模拟游戏，玩家通过精心规划的活动如种植作物、采矿、钓鱼和制作工艺品来恢复和扩展农场。玩家可以与村民建立关系，参与季节性活动，揭开山谷的秘密。游戏鼓励策略规划和时间管理，每一天都带来新的机遇和挑战。玩家需平衡精力和资源，以最大化农场的生产力和盈利能力。
- Dealer's Life 2, a simulation game where players manage a pawn shop. They must assess the value of items, haggle with customers, and make strategic decisions to grow their business. The game offers a dynamic market influenced by trends, customer preferences, and random events, requiring players to adapt and refine their negotiation tactics.  
• 《当铺人生2》 (Dealer's Life 2)，一款模拟经营游戏，玩家管理一家当铺。需评估物品价值，与顾客讨价还价，并做出战略决策以发展业务。游戏市场动态受趋势、顾客偏好和随机事件影响，要求玩家不断调整和优化谈判策略。
- Cities: Skylines, a 3D, top-down view, city-building game where players take on the role of a city mayor, tasked with the development and management of a thriving metropolis, engaging in urban planning by controlling zoning, road placement, taxation, public services, and public transportation in an area. They must balance the needs and desires of the population with the city's budget,  
• 《城市：天际线》 (Cities: Skylines)，一款3D俯视视角的城市建设游戏，玩家扮演市长，负责发展和管理繁荣的大都市，通过控制分区、道路布局、税收、公共服务和公共交通进行城市规划。玩家必须平衡居民需求与城市预算，

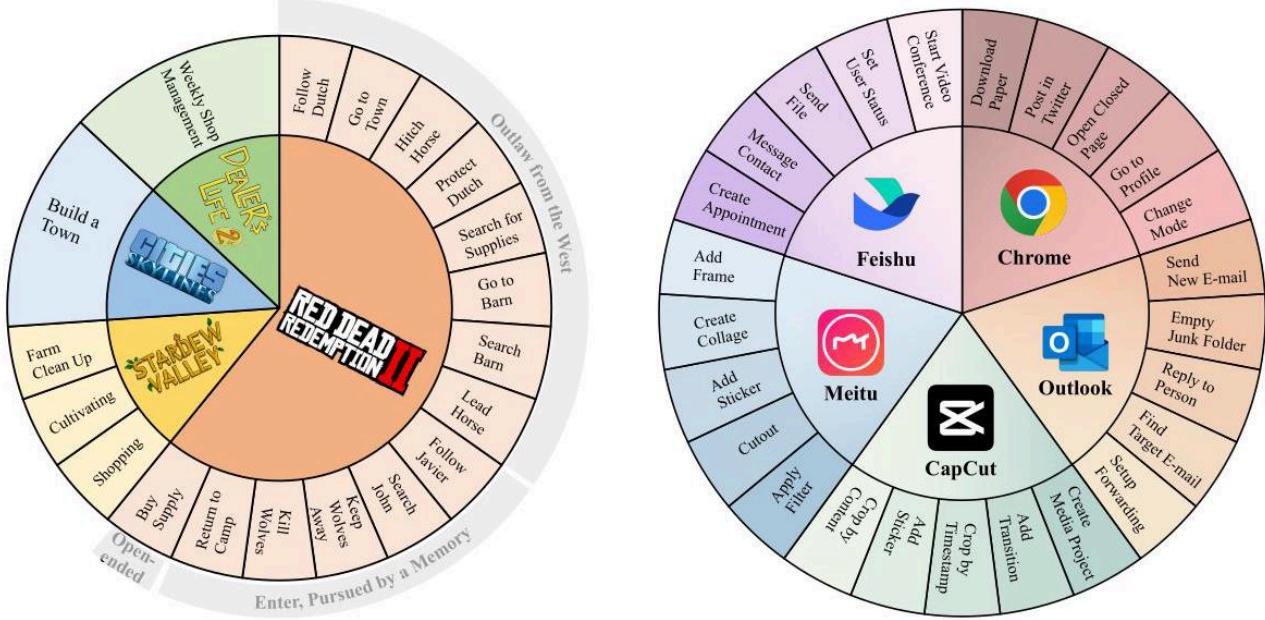


Figure 3: Overview of all game tasks (left) in RDR2, Stardew Valley, Cities: Skylines, and Dealer's Life 2 and application tasks (right) in Chrome, Outlook, CapCut, Meitu, and Feishu.

图3：左侧为RDR2、星露谷物语、城市：天际线和当铺人生2中所有游戏任务的概览，右侧为Chrome、Outlook、CapCut、美图和飞书中的应用任务概览。

addressing issues such as traffic congestion, pollution, and citizen satisfaction. The game provides a sandbox environment where creativity and strategic thinking are key to building efficient and aesthetically pleasing urban landscapes. It also requires highly precise mouse control.

解决交通拥堵、污染和市民满意度等问题。游戏提供了一个沙盒环境，创造力和战略思维是构建高效且美观城市景观的关键，同时对鼠标操作的精确度要求极高。

The target set of diverse software applications for evaluation includes: Chrome, Outlook, CapCut, Meitu, and Feishu, as well as one comprehensive software benchmark, OSWorld [66].

评估所选的多样化软件应用包括：Chrome、Outlook、CapCut、美图和飞书，以及一个综合软件基准测试OSWorld [66]。

As shown in Figure 3, for each game and software application, representative tasks are designed to measure the various abilities of the agent comprehensively. Experimental results show that CRADLE exhibits remarkable generalization ability and impressive performance across the four previously unexplored commercial video games, the five target software applications, and the comprehensive contemporaneous OSWorld benchmark. To our best knowledge, CRADLE is the first to enable LMM-based agents to follow the main storyline and complete 40-minute-long real missions in a complex AAA game, RDR2. CRADLE also manages to create a city of a thousand people in Cities: Skylines, farm and harvest parsnips in Stardew Valley, trade and bargain with a maximal weekly total profit of 87% in Dealer's Life 2. Besides, CRADLE can not only operate daily software, like Chrome and Outlook, but also edit images and videos using Meitu and CapCut, and perform office tasks in Feishu. Able to interact with software in a unified manner, CRADLE greatly extends the reach of AI agents by making it easy to convert any software, especially complex games, into benchmarks to evaluate agents' various abilities and facilitate further data collection, paving the way for generalism. We hope the open-source CRADLE framework and its holistic evaluation protocol, i.e., tasks and metrics, on various environments, can accelerate the development of more powerful foundation agents, thereby advancing the path towards AGI.

如图3所示，对于每款游戏和软件应用，设计了代表性任务以全面衡量智能体的多种能力。实验结果表明，CRADLE在四款此前未被探索的商业视频游戏、五款目标软件应用及综合的当代OSWorld基准测试中展现出卓越的泛化能力和出色的性能。据我们所知，CRADLE是首个使基于大语言模型（LMM）的智能体能够遵循主线剧情并完成复杂AAA

游戏RDR2中长达40分钟真实任务的系统。CRADLE还成功在城市：天际线中建造了千人城市，在星露谷物语中种植和收获欧防风，在当铺人生2中实现了最高87%的周总利润交易和讨价还价。此外，CRADLE不仅能操作日常软件如Chrome和Outlook，还能使用美图和CapCut编辑图像和视频，并在飞书中执行办公任务。通过统一方式与软件交互，CRADLE极大地扩展了AI智能体的应用范围，使任何软件，尤其是复杂游戏，易于转化为评估智能体多种能力的基准并促进进一步数据收集，为通用智能铺平道路。我们希望开源的CRADLE框架及其在多种环境下的整体评估协议（即任务和指标）能加速更强大基础智能体的发展，推动通向通用人工智能（AGI）的进程。

## 5 2 Related Work

### 6 2 相关工作

#### 6.1 2.1 Environments and Benchmarks for Computer Control

##### 6.2 2.1 计算机控制的环境与基准

Environments and Benchmarks on Software Applications. Simulated environments on computers have been popular benchmarks and testbeds for the research community. Earlier computer control environments primarily focused on web navigation tasks [13, 30, 34, 52, 70, 76]. Recent benchmarks start to include various common software [27, 66], aiming to develop a generalist agent in the digital world. However, none of them takes video games into consideration, missing a key component of computer control.

软件应用的环境与基准。计算机上的模拟环境一直是研究社区广泛使用的基准和测试平台。早期的计算机控制环境主要聚焦于网页导航任务[13, 30, 34, 52, 70, 76]。近期的基准开始涵盖各种常见软件[27, 66]，旨在开发数字世界中的通用智能体。然而，尚无研究将视频游戏纳入考虑，忽视了计算机控制的关键组成部分。

Environments and Benchmarks on Video Games. On the other side, many research environments are built on top of video games, significantly advancing the study of decision-making, especially, reinforcement learning (RL).

Examples include but are not limited to Atari games [4], Super Mario Bros [28], Google Research Football [31], Minecraft [16, 21, 26], Dota II [5], StarCraft II [15, 50, 56], Quake III [25], Gran Turismo [64], Diplomacy [3] and Civilization [47]. Additionally, many custom-built environments, especially grid world and embodied scenarios, are created from scratch in a game-like manner to facilitate agent development, such as BabyAI [11], Melting Pot [32], Overcooked [9, 62, 65], VRKitchen [19], VirtualHome [45], iGibson [33, 51], ProcTHOR [12], Habitat [37, 46, 54], and Generative agents [44].

视频游戏的环境与基准。另一方面，许多研究环境构建于视频游戏之上，显著推动了决策制定，尤其是强化学习（Reinforcement Learning, RL）的研究。示例包括但不限于Atari游戏[4]、超级马里奥兄弟[28]、谷歌研究足球[31]、Minecraft[16, 21, 26]、Dota II[5]、星际争霸II[15, 50, 56]、Quake III[25]、Gran Turismo[64]、Diplomacy[3]和文明帝国[47]。此外，许多定制环境，尤其是网格世界和具身场景，以游戏化方式从零开始创建以促进智能体开发，如BabyAI[11]、Melting Pot[32]、Overcooked[9, 62, 65]、VRKitchen[19]、VirtualHome[45]、iGibson[33, 51]、ProcTHOR[12]、Habitat[37, 46, 54]和生成式智能体[44]。

Each of these environments highly relies on the accessibility of the open-source code or provided built-in APIs. Significant human efforts are required for implementation and encapsulation, enabling agent interaction. Therefore, despite the abundance of software and games available for human use, only a limited number are accessible to agents, especially for commercial closed-source games and software applications. Additionally, the lack of consensus on environment standards further complicates the interaction, as each environment has specific observation and action spaces, tailored to its unique requirements. This variation exacerbates the challenge of enabling agents to interact with diverse environments and collect data with a consistent level of fine-grained semantics to improve the agent's capabilities. Few agents can complete tasks across multiple environments so far.

这些环境高度依赖开源代码的可访问性或提供的内置API。实现和封装需要大量人力投入，以支持智能体的交互。因此，尽管人类可用的软件和游戏丰富，但可供智能体访问的数量有限，尤其是商业闭源游戏和软件应用。此外，环境标准缺乏共识进一步加剧了交互的复杂性，因为每个环境都有特定的观测空间和动作空间，针对其独特需求量身定

制。这种差异加大了智能体与多样环境交互并以一致的细粒度语义收集数据以提升能力的难度。目前，能够跨多个环境完成任务的智能体寥寥无几。

Similar to OpenAI Universe [41] and SIMA [48], our goal is to explore a unified way that allows agents to interact for measuring and training agents' abilities across a wide range of games, websites, and other applications without heavy human efforts needed. This approach aims to prove that diverse software applications and games can serve as out-of-the-box environments for AI development.

类似于OpenAI Universe[41]和SIMA[48]，我们的目标是探索一种统一方式，使智能体能够在广泛的游戏、网站及其他应用中进行交互，以衡量和训练智能体能力，且无需大量人力投入。该方法旨在证明多样的软件应用和游戏可以作为开箱即用的AI开发环境。

### 6.3 2.2 LMM-based Agents for Computer Tasks

#### 6.4 2.2 基于大语言模型（LMM）的计算机任务智能体

Agents for Software Manipulation. Agents for software applications are developed to complete tasks such as web navigation [13, 38, 76] and software application control [27, 49, 69]. While previous LLM-based web agents [13, 20, 75, 76] show some promising results in effectively interacting with content on webpages, they usually use raw HTML code and DOM tree as input and interact with the available element IDs, ignoring the rich visual patterns with key information, like icons, images, and spatial relations. Recently, multimodal web agents [18, 22, 40, 63, 67, 73, 74] and mobile app agents [58, 69] have been explored. Though using screenshots as input, they still rely on built-in APIs and advanced tools to get internal information, like available interactive element IDs, to execute corresponding actions, which greatly limits their applicability. Other train-based agents [10, 17, 23] also suffer from generalizing to unseen software and tasks. Moreover, all of these works primarily focus on static websites and software, which greatly reduces the need for timeliness and simplifies the setting by ignoring the dynamics between adjacent screenshots, i.e., animations, and incomplete action space without considering the duration of the key press and different mouse mode. It results in the failure of deployment to the tasks with rapid graphics changes, e.g., game playing.

软件操作智能体。针对软件应用的智能体被开发用于完成诸如网页导航[13, 38, 76]和软件应用控制[27, 49, 69]等任务。尽管先前基于大语言模型（LLM）的网页智能体[13, 20, 75, 76]在有效交互网页内容方面展现出一定潜力，但它们通常使用原始HTML代码和DOM树作为输入，并通过可用元素ID进行交互，忽视了包含关键信息的丰富视觉模式，如图标、图像和空间关系。近期，多模态网页智能体[18, 22, 40, 63, 67, 73, 74]和移动应用智能体[58, 69]被探索。尽管使用截图作为输入，它们仍依赖内置API和高级工具获取内部信息，如可交互元素ID，以执行相应操作，这极大限制了其适用性。其他基于训练的智能体[10, 17, 23]也存在对未见软件和任务泛化能力不足的问题。此外，所有这些工作主要聚焦于静态网站和软件，极大降低了对时效性的需求，并通过忽略相邻截图间的动态变化（如动画）及未考虑按键持续时间和不同鼠标模式的不完整动作空间，简化了设置。这导致其无法部署于图形快速变化的任务，如游戏操作。

Agents for Game Playing. Several attempts try to develop foundation agents for complex video games, such as Minecraft [57, 60, 61], Starcraft II [36] and Civilization-like game [47] with textual observations obtained from internal APIs and pre-defined semantic actions. Although JARVIS-1 [60] claims to interact with the environment in a human-like manner with the screenshots as input and mouse and keyboard for control, its action space is predefined as a hybrid space composed of keyboard, mouse, and API. The game-specific observation and action spaces prohibit the generalization of them to other novel games. Pre-trained with videos with action labels, VPT [2] manages to output mouse and keyboard control with raw screenshots as input without any additional information. However, collecting videos with action labels is time-consuming and costly, which is difficult to generalize to multiple environments. Another concurrent work, SIMA [48] trained embodied agents to complete 10-second-long tasks over ten 3D video games. Though their results are promising to scale up, they focus on behavior cloning with gameplay data from human experts, resulting in a high expense.

游戏操作智能体。已有若干尝试开发复杂视频游戏的基础智能体，如Minecraft[57, 60, 61]、星际争霸II[36]和类似文明帝国的游戏[47]，其文本观测来自内部API，动作为预定义语义动作。尽管JARVIS-1[60]声称以人类方式交互环境，输入为截图，控制通过鼠标和键盘，其动作空间预定义为键盘、鼠标和API组成的混合空间。游戏特定的观测和

动作空间限制了其向其他新游戏的泛化能力。通过带动作标签的视频预训练，VPT[2]能够以原始截图为输入输出鼠标和键盘控制，无需额外信息。然而，收集带动作标签的视频耗时且成本高，难以泛化至多环境。另一同期工作SIMA[48]训练具身智能体完成十个3D视频游戏中时长10秒的任务。尽管其结果有望扩展，但其侧重于利用人类专家的游戏数据进行行为克隆，成本较高。

In both targeting complex video games and diverse software applications, CRADLE attempts to explore a new way to efficiently interact with different complex environments in a unified manner and facilitate further data collection. In a nutshell, to our best knowledge, there are currently no agents under the GCC setting, reported to show superior performance and generalization in complex video games and across computer tasks. In this work, we make a preliminary attempt to explore and benchmark diverse environments in this setting, applying our framework to diverse challenging environments under GCC and proposing an approach where any software can be used to benchmark agentic capabilities in it.

在针对复杂视频游戏和多样化软件应用的目标中，CRADLE尝试探索一种以统一方式高效交互不同复杂环境并促进进一步数据收集的新方法。简而言之，据我们所知，目前尚无在GCC（通用计算控制）设置下，报告显示在复杂视频游戏和跨计算机任务中表现优异且具备良好泛化能力的智能体。在本工作中，我们初步尝试探索并基准测试该设置下的多样环境，将我们的框架应用于GCC下的多种挑战性环境，并提出一种方法，使任何软件都可用于基准测试智能体能力。

## 7 3 The CRADLE Framework

### 8 3 CRADLE框架

To pursue GCC, we propose CRADLE, illustrated in Figure 4, a modular and flexible LMM-powered framework that can properly handle the challenges GCC presents. The framework should have the ability to understand and interpret computer screens and dynamic changes between consecutive frames from arbitrary software and be able to generate reasonable computer control actions to be executed precisely. This suggests that a multimodal model with powerful vision and reasoning capabilities, in addition to rich knowledge of computer UI and control, is a requirement. In this work, we leverage GPT-4o [43] as the framework's backbone model.

为实现GCC，我们提出了CRADLE，如图4所示，这是一个模块化且灵活的基于大型多模态模型（LMM）的框架，能够妥善应对GCC带来的挑战。该框架应具备理解和解析任意软件的计算机屏幕及连续帧间动态变化的能力，并能生成合理的计算机控制操作以精确执行。这意味着需要一个具备强大视觉和推理能力、以及丰富计算机用户界面（UI）和控制知识的多模态模型。在本工作中，我们采用GPT-4o [43]作为框架的主干模型。

#### 8.1 3.1 Environment IO

##### 8.2 3.1 环境输入输出

Observation and Action Space. CRADLE only takes a video clip, recording the execution of the last action, as input and outputs keyboard and mouse operations to interact with environments. The observation space is made up of complete screen videos with different lengths. For the action space, it includes all possible keyboard and mouse operations, including key\_press, key\_hold, key\_release, mouse\_move, and wheel\_scroll. These operations can be combined in various ways to form combos and shortcuts, execute rapid key sequences, or coordinate timings. We choose to use Python code to simulate these operations and encapsulate them into an io\_env class. Note that seldom previous work takes key\_hold, key\_release and mouse moving speed into consideration, which are critical in games (e.g., RDR2, for opening weapon wheel and changing view). The

观察与动作空间。CRADLE仅以记录上一次操作执行过程的视频片段作为输入，输出键盘和鼠标操作以与环境交互。观察空间由不同长度的完整屏幕视频组成。动作空间包括所有可能的键盘和鼠标操作，如按键按下（key\_press）、按键保持（key\_hold）、按键释放（key\_release）、鼠标移动（mouse\_move）和滚轮滚动（wheel\_scroll）。这些操作可组合形成连招和快捷键，执行快速按键序列或协调时序。我们选择使用Python代码模拟这些操作，并封装为io\_env类。值得注意的是，之前很少有工作考虑按键保持、按键释放和鼠标移动速度，这些在游戏中（如《荒野大镖客2》（RDR2）用于打开武器轮盘和视角切换）至关重要。

<sup>1</sup> Keys include both keyboard keys and mouse buttons.

<sup>1</sup> 键包括键盘按键和鼠标按钮。

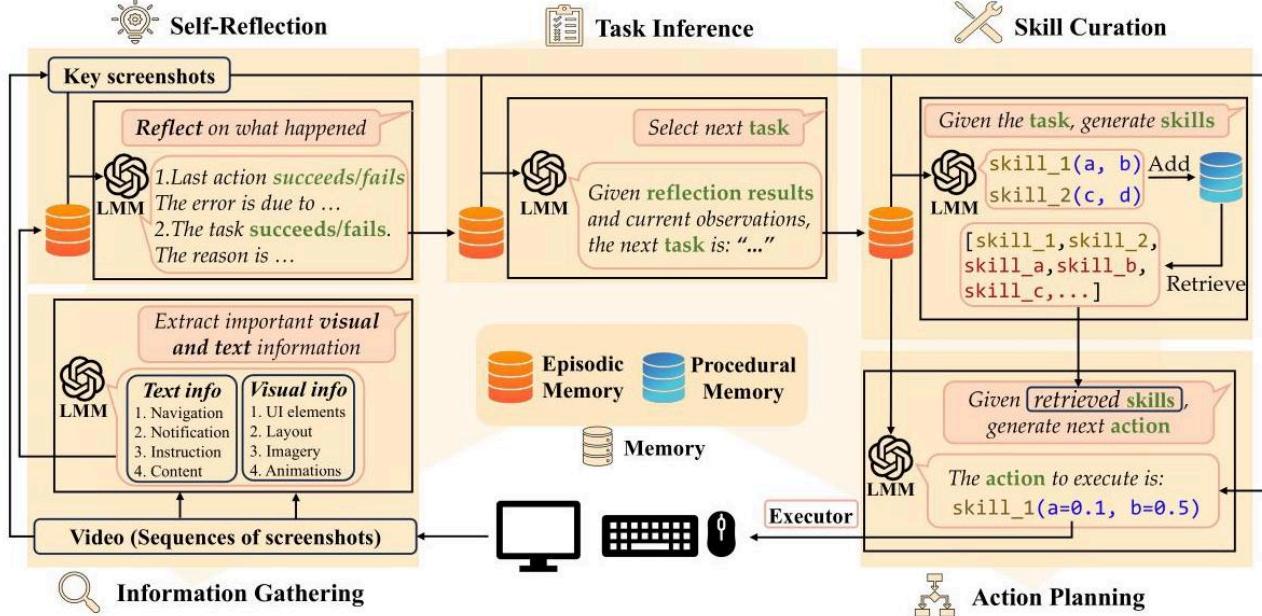


Figure 4: An overview of the CRADLE framework. CRADLE takes video from the computer screen as input and outputs computer keyboard and mouse control determined through inner reasoning.

图4：CRADLE框架概览。CRADLE以计算机屏幕视频为输入，通过内部推理决定计算机键盘和鼠标控制输出。

asynchronization brought by these temporal-extended actions introduces additional challenges for the control. These time extension actions bring extra challenges for control.

**Information Gathering.** Provided with a video clip as input, it is critical for CRADLE to capture and extract all useful visual and textual information to understand the recent situation and perform further reasoning. Visual information includes layout, imagery, animations, and UI elements which pose high spatial perception and visual understanding requirements for LMM models. Moreover, we depend on their OCR capabilities to extract textual information in images, which usually includes content (headings and paragraphs), navigation labels (menus and links), notifications, and instructions to convey messages and guide users. Moreover, for each environment, we enhance LMMs' abilities with different tools such as template matching [8], Grounding DINO [35], and SAM [29] to provide additional grounding for object detection and localization.

信息采集。以视频片段为输入，CRADLE必须捕捉并提取所有有用的视觉和文本信息，以理解当前情境并进行进一步推理。视觉信息包括布局、图像、动画和UI元素，这对LMM模型提出了较高的空间感知和视觉理解要求。此外，我们依赖其光学字符识别（OCR）能力提取图像中的文本信息，通常包括内容（标题和段落）、导航标签（菜单和链接）、通知和指令，用以传递信息和引导用户。针对每个环境，我们还通过模板匹配[8]、Grounding DINO[35]和SAM[29]等工具增强LMM的能力，为目标检测和定位提供额外支撑。

Skill and Action Generation As shown in Figure 5, to bridge the gap between semantic actions generated by LMMs and OS-level executable actions, CRADLE uses LMMs to generate code functions as semantic-level skills, which encapsulate lower-level keyboard and mouse control. Similarly to how humans improve while playing, these skills can be developed from scratch according to in-game tutorials and guidance, game manuals and settings, or through self-exploration as the game progresses. These skills can also be pre-defined or composited to solve more complex tasks. An action usually consists of a single or multiple skills instantiated with any necessary parametric aspects, such as duration, position, and speed.

技能与动作生成。如图5所示，为弥合LMM生成的语义动作与操作系统级可执行动作之间的差距，CRADLE使用LMM生成代码函数作为语义层面的技能，封装底层键盘和鼠标控制。类似人类在游戏中通过教程、游戏手册和设置或自我探索逐步提升技能，这些技能可从零开始开发，也可预定义或组合以解决更复杂任务。一个动作通常由一个或多个技能组成，并带有必要的参数，如持续时间、位置和速度。

Action Execution. After CRADLE generates actions and decides to execute them in the environment, an Executor is then triggered to map these semantic actions to the OS-level keyboard and mouse commands to interact with the environment.

动作执行。在CRADLE生成动作并决定执行后，执行器（Executor）被触发，将这些语义动作映射为操作系统级的键盘和鼠标命令，以与环境交互。



Figure 5: Examples for skill generation according to in-game guidance in RDR2 (left), in-game manual in Stardew Valley (middle), self-exploration in Cities: Skylines (right). Code and comments are shown in brevity.

图5：根据《荒野大镖客2》（左）游戏内指导、《星露谷物语》（中）游戏内手册和《城市：天际线》（右）自我探索生成技能的示例。代码和注释简略展示。

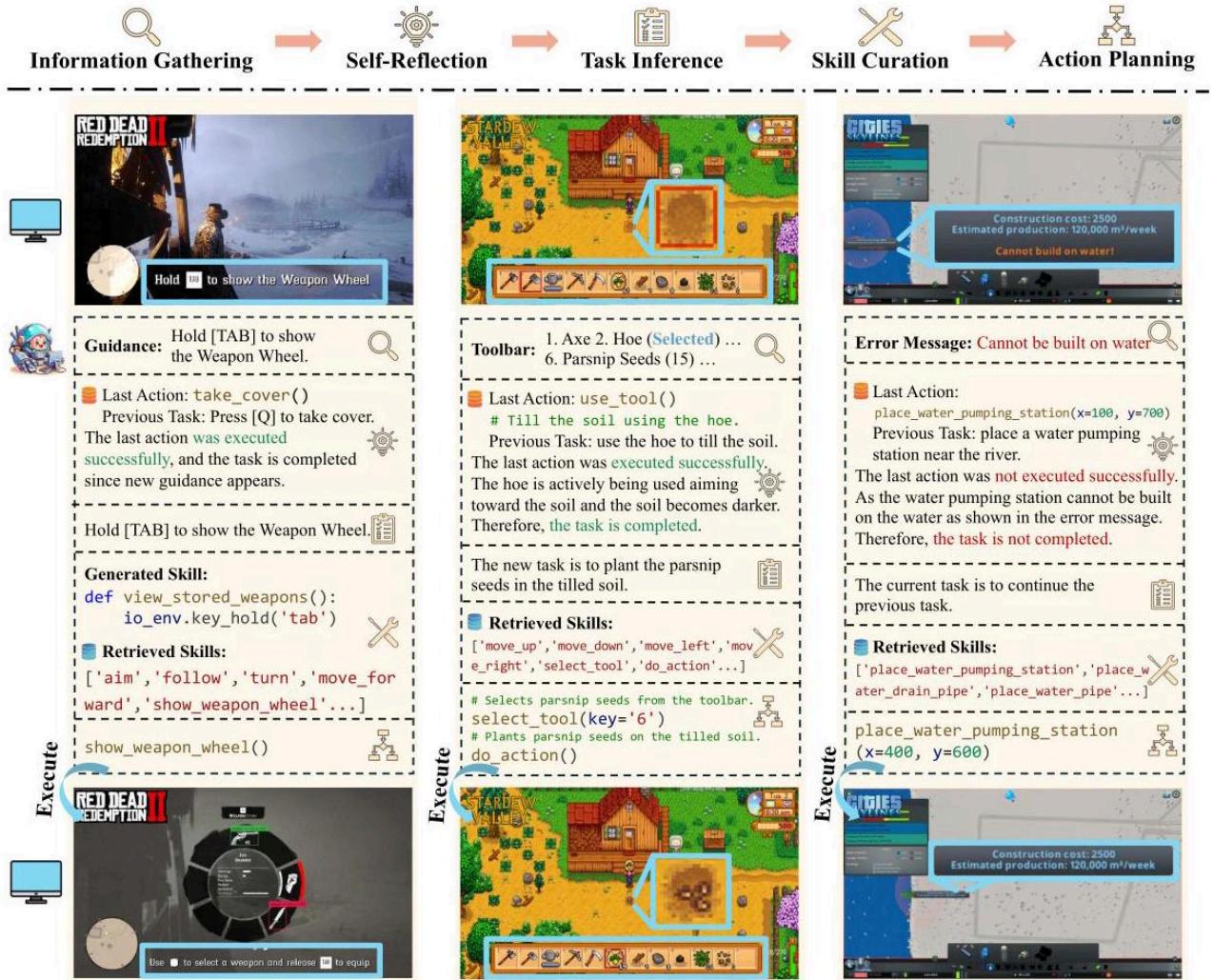


Figure 6: Illustrative examples of CRADLE's complete workflow in RDR2 (left), Stardew Valley (middle) and Cities: Skylines (right). Prompts are shown partially for brevity.

图6: CRADLE在《荒野大镖客2》（左）、《星露谷物语》（中）和《城市：天际线》（右）中的完整工作流程示例。提示部分简略展示。

### 8.3 3.2 Memory

### 8.4 3.2 记忆

CRADLE stores and maintains all the useful information from the environment or outputted by each module through a memory mechanism, consisting of episodic memory and procedural memory.

CRADLE通过记忆机制存储和维护来自环境或各模块输出的所有有用信息，该机制包括情景记忆和程序记忆。

**Episodic Memory.** Episodic memory is used to maintain current and past experiences, including key screenshots from each video observation, and everything useful outputted by LMMs and advanced tools, e.g., textual and visual information, actions, tasks, and reasoning from each module. To facilitate retrieval and storage, periodical summarization is conducted to abstract recently added multimodal information into long-term summaries. The incorporation of episodic memory enables CRADLE to effectively retain crucial information over extended periods.

情景记忆。情景记忆用于维护当前和过去的经历，包括每个视频观察的关键截图，以及由大规模多模态模型

(LMMs) 和高级工具输出的所有有用信息，例如文本和视觉信息、动作、任务以及各模块的推理。为了便于检索和存储，会定期进行总结，将最近添加的多模态信息抽象为长期摘要。情景记忆的引入使CRADLE能够有效地在较长时间内保留关键信息。

Procedural Memory. This memory is specific to storing and retrieving skills in code form, which can be learned from scratch, as shown in Figure 5, or pre-defined in procedural memory. Skills can be added, updated, or composed in the procedural memory. The most relevant skills for a given task and situation will be retrieved to support action planning, therefore, as CRADLE continuously acquires new skills during interactions, it is critical that this memory can effectively calculate skill relevance.

程序性记忆。此类记忆专门用于以代码形式存储和检索技能，这些技能可以从零开始学习，如图5所示，或预先定义在程序性记忆中。技能可以在程序性记忆中被添加、更新或组合。为了支持行动规划，将检索与特定任务和情境最相关的技能，因此，随着CRADLE在交互过程中不断获取新技能，确保该记忆能够有效计算技能相关性至关重要。

## 8.5 3.3 Reasoning

### 8.6 3.3 推理

Based on the extracted information from the video observations and relevant information from its memory, CRADLE needs to do high-level reasoning and then make the next decision. This process is analogous to "reflect on the past, summarize the present, and plan for the future", which is broken down into the following modules.

基于从视频观察中提取的信息及其记忆中的相关信息，CRADLE需要进行高级推理，然后做出下一步决策。这个过程类似于“回顾过去，总结现在，规划未来”，并分解为以下几个模块。

Self-Reflection. The reflection module initially evaluates whether the last executed action was successfully carried out and whether the task was completed. Sequential key screenshots from the last video observation, along with the previous context for action planning and task inference are fed to the LMM for reasoning. Additionally, we also request the LMM to provide an analysis of any failure. This valuable information enables CRADLE to try and remedy inappropriate decisions or less-than-ideal actions. Furthermore, reflection can also be leveraged to inform re-planning of the

自我反思。反思模块首先评估上一次执行的动作是否成功完成以及任务是否达成。来自最后一次视频观察的关键连续截图，以及用于动作规划和任务推断的先前上下文，被输入到大规模语言模型（LMM）进行推理。此外，我们还要求LMM对任何失败进行分析。这些宝贵的信息使CRADLE能够尝试纠正不当决策或不理想的动作。此外，反思还可以用于指导重新规划

task and bring the agent closer to target task completion, better understand the factors that led to previous successes, or suggest how to update or improve specific skills.

任务并使代理更接近目标任务的完成，更好地理解导致以往成功的因素，或建议如何更新或改进特定技能。

Task Inference. After reflecting on the outcome of the last executed action, CRADLE needs to analyze the current situation to infer the most suitable task for the current moment. We let LMMs estimate the highest priority task to perform and when to stop an ongoing task and start a new one.

任务推断。在反思上一次执行动作的结果后，CRADLE需要分析当前情况，以推断当前时刻最合适的任务。我们让大规模语言模型（LMMs）评估应执行的最高优先级任务，以及何时停止正在进行的任务并开始新任务。

Skill Curation. As the task is determined, CRADLE needs to prepare the tactics to accomplish it, by retrieving useful skills from the procedural memory, updating skills, or generating new ones. The new skill will be stored in the procedural memory for future utilization. Action Planning. CRADLE needs to select the appropriate skills from the curated skill set and instantiate these skills into a sequence of executable actions by specifying any necessary parametric aspects (e.g., duration, position, and target) according to the current task and history information. The generated action is then fed to the Executor for interaction with the environment.

技能策划。任务确定后，CRADLE需要准备完成任务的策略，通过从程序记忆中检索有用技能、更新技能或生成新技能。新技能将被存储在程序记忆中以备将来使用。行动规划。CRADLE需要从策划的技能集中选择合适的技能，并根据当前任务和历史信息，通过指定必要的参数（如持续时间、位置和目标）将这些技能实例化为一系列可执行的动作。生成的动作随后被传递给执行器以与环境交互。

Through these six modules, the input video is processed in stages: first into reasoning, then into semantic skills and actions, and finally into low-level keyboard and mouse operations. This comprehensive conversion covers all essential interactive data needed for both high-level planning and low-level control, enabling a unified approach to data collection for further self-improvement.

通过这六个模块，输入视频被分阶段处理：首先进行推理，然后转化为语义技能和动作，最后转化为低级的键盘和鼠标操作。这一全面的转换涵盖了高层规划和低层控制所需的所有关键交互数据，实现了统一的数据采集方法，便于后续的自我提升。

## 9 4 Empirical Studies

### 10 4 实证研究

In this section, we report empirical results of applying CRADLE in various challenging environments representative of GCC setting to demonstrate CRADLE's capabilities in decision-making, UI understanding, and manipulation<sup>2</sup>. To facilitate reproducibility, we also provide setup and load scripts (e.g., game saves and checkpoints) in the code repository to reset state for task execution where appropriate.

本节中，我们报告了在代表GCC环境的各种挑战性场景中应用CRADLE的实证结果，以展示CRADLE在决策、用户界面理解和操作方面的能力<sup>2</sup>。为便于复现，我们还在代码库中提供了设置和加载脚本（例如游戏存档和检查点），以便在适当情况下重置任务执行状态。

#### 10.1 4.1 Experimental Settings

#### 10.2 4.1 实验设置

Here we provide a brief introduction to our experimental settings. More implementation details, environment and task descriptions, and prompts used can be found in Appendices A to H.

这里我们简要介绍我们的实验设置。更多的实现细节、环境和任务描述以及使用的提示可见附录A至H。

Implementation Details. To lower the frequency of interaction with the backbone model, video observation is recorded at 2 fps, which proves sufficient for information gathering without missing any important information in most situations. If not specifically mentioned, all experiments are conducted in five runs under a maximum step limit, using OpenAI's latest model, gpt-4o-2024-05-13 [43]. Same as Voyager [57], we use OpenAI's text-embedding-ada-002 model [42] to generate embeddings for each skill, stored in the procedural memory and retrieved according to the similarities. It is important to note that, due to the dynamism of the RDR2 and Stardew Valley and the LMM inference and communication latency, we must pause those game environments while waiting for backbone model responses. Other environments execute continuously. All software and games can be run on regular Windows 10 machines, except for RDR2, which is tested on two machines with an NVIDIA RTX-3060 GPU and RTX-4090 GPU separately. We observe a slight performance gain due to the stability of RTX-4090 GPU during the gameplay. 实现细节。为了降低与主干模型的交互频率，视频观察以2帧每秒的速度记录，这在大多数情况下足以收集信息且不会遗漏重要内容。除非特别说明，所有实验均在最大步数限制下进行五次运行，使用OpenAI最新模型gpt-4o-2024-05-13 [43]。与Voyager [57]相同，我们使用OpenAI的text-embedding-ada-002模型 [42]为每项技能生成嵌入，存储在程序记忆中并根据相似度进行检索。需要注意的是，由于《荒野大镖客2》（RDR2）和《星露谷物语》（Stardew Valley）的动态性以及大语言模型（LMM）推理和通信延迟，我们必须在等待主干模型响应时暂停这些游戏环境。其他环境则持续运行。除RDR2外，所有软件和游戏均可在普通Windows 10机器上运行，RDR2分别在配备NVIDIA RTX-3060 GPU和RTX-4090 GPU的两台机器上测试。我们观察到RTX-4090 GPU在游戏过程中因其稳定性带来轻微的性能提升。

Evaluation Methods. Unlike conventional research benchmarks, which usually provide grounding signals for evaluation, it is difficult to have a unified and general method to determine whether a task is completed automatically in diverse software, especially in video games. Similarly to SIMA [48], we apply human evaluation to all tasks across application software and games. Moreover, to provide more quantitative results and a comparison

baseline, we provide results for the OSWorld [66] benchmark, a contemporaneous benchmark that provides evaluation scripts for at least one solution per task.

评估方法。与通常提供评估依据信号的传统研究基准不同，在多样化的软件中，尤其是视频游戏中，很难有统一且通用的方法自动判断任务是否完成。类似于SIMA [48]，我们对所有应用软件和游戏中的任务均采用人工评估。此外，为了提供更量化的结果和比较基线，我们提供了OSWorld [66]基准的结果，该基准是一个同期基准，为每个任务至少提供一个解决方案的评估脚本。

Task Introduction. For RDR2, we mainly focus on evaluating agents on the first two missions of the main storyline in Chapter I, which can be divided into 13 tasks according to the in-game checkpoints, which include but are not limited to NPC following, house exploration, and combat. A novice player typically takes about 40 minutes to complete these missions. Few studies tackle such long-duration tasks and rich semantic environments. It is an ideal scenario to emulate a new human player learning to play the game from scratch according to the rich in-game tutorials and hints. Despite the missions in the main storyline, we also designed an open-ended task, Buy Supply, in the open-ended world, Chapter II, where the agent is instructed to go to the General Store in Valentine town from the camp

任务介绍。对于《荒野大镖客救赎2》（Red Dead Redemption 2，简称RDR2），我们主要关注评估代理在第一章主线的前两个任务，这些任务根据游戏内检查点可分为13个子任务，包括但不限于NPC跟随、房屋探索和战斗。新手玩家通常需要约40分钟完成这些任务。很少有研究涉及如此长时间的任务和丰富的语义环境。这是一个理想场景，用以模拟新玩家根据丰富的游戏内教程和提示从零开始学习游戏。除了主线任务外，我们还设计了一个开放式任务“购买补给”，位于第二章的开放世界中，指示代理从营地前往瓦伦丁镇的杂货店购买补给。

---

<sup>2</sup> Result tables present two main indicator formats:  $\mu \pm \sigma$ , representing mean and standard deviation of five runs; and  $(s/t)$  representing  $s$  successful runs out of a total of  $t$  runs.

<sup>2</sup> 结果表展示两种主要指标格式： $\mu \pm \sigma$ ，表示五次运行的均值和标准差；以及 $(s/t)$ ，表示在总共 $t$ 次运行中成功的 $s$ 次。

---

for supplementary supply. For this open-ended task, seldom in-game guidance will appear. The agent needs to analyze and propose feasible solutions to complete the mission. For Stardew Valley, we propose three essential tasks at the stage of the game, i.e., Farm Cleanup: Clear the obstacles on the farm, such as weeds, stones, and trees, as much as possible to prepare for farming; 2) Cultivation: Plant the parsnip seed, water every day and harvest at least one parsnip; 3) Shopping: Go to the general store in the town, which is out of the scope of the current map, to buy more seeds and return home. For Dealer's Life, the agent is tasked with managing a shop for a week, appraising item values and haggling with the customers to secure deals. For Cities: Skylines, the task is to build a reasonable city ending in as much population as possible, with the initial starting funds of C70,000, and basic road, water and power facilities. Moreover, we define five representative domain-specific tasks for each of the five Software Applications in our diverse target set.

用于补给的开放式任务中，几乎没有游戏内指导出现。代理需要分析并提出可行方案以完成任务。对于《星露谷物语》（Stardew Valley），我们提出了游戏阶段的三个核心任务：1）农场清理：尽可能清除农场上的障碍物，如杂草、石头和树木，为耕种做准备；2）种植：种植欧防风种子，每天浇水并至少收获一颗欧防风；3）购物：前往镇上的杂货店（当前地图范围之外）购买更多种子并返回家中。对于《商人生活》（Dealer's Life），代理需管理一家商店一周，评估物品价值并与顾客讨价还价以达成交易。对于《城市：天际线》（Cities: Skylines），任务是以7万货币的初始资金和基本的道路、水电设施，建设一个合理的城市，尽可能达到最大人口。此外，我们为多样目标集中的五款软件应用各定义了五个具有代表性的领域特定任务。

## 10.3 4.2 Performance across Tasks and Environments

### 10.4 4.2 任务与环境间的性能表现

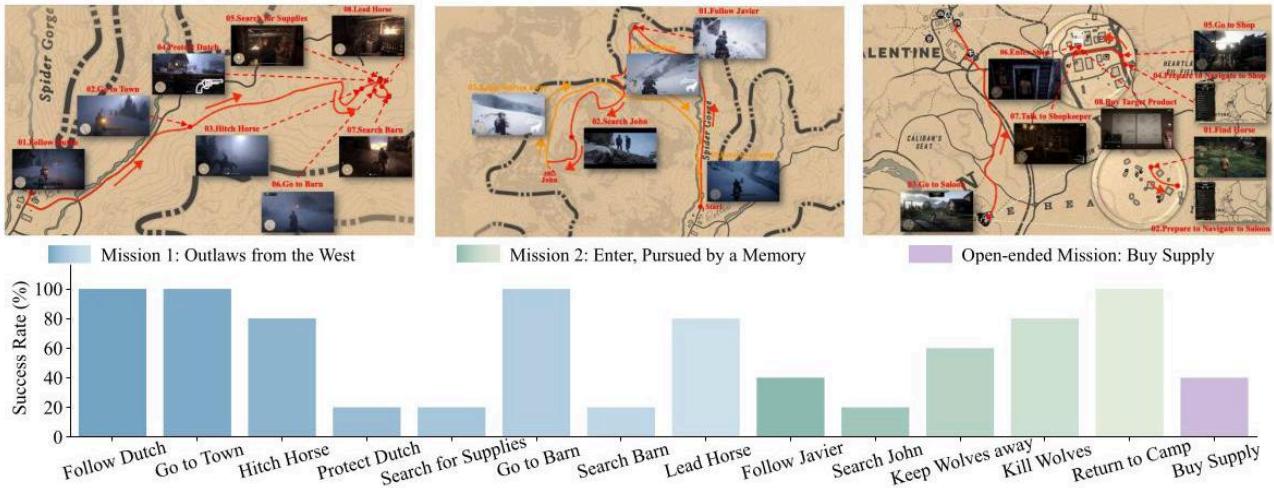


Figure 7: Trajectory and success rates of 13 main storyline tasks and 1 open-world task in RDR2. Each task is run with a maximum of 500 steps. Agents can retry the checkpoint if the task fails or the character dies.

图7: 《荒野大镖客救赎2》中13个主线任务和1个开放世界任务的轨迹及成功率。每个任务最多运行500步。任务失败或角色死亡时，代理可重试检查点。

Red Dead Redemption 2. As shown in Figure 7, CRADLE can achieve a high success rate in simple tasks like following an NPC or going to specific locations on the ground (e.g., Follow Dutch and Go to Barn). Another following task, Follow Javier, and the searching task, Search John, are dangerous for the rugged and winding path up to the snow mountain with cliffs. In addition, GPT-4o struggles with real-time combat tasks and searching tasks due to its inability to accurately locate enemies or objects and precisely time decisions. Even equipped with additional detection tools, like Grounding DINO [35], the success rate drops significantly to 20% in the task of Protect Dutch, which requires nighttime combat. Additionally, indoor tasks like Search for Supplies and Search Barn are also challenging due to GPT4-o's poor spatial perception, which finds it difficult to locate target objects and ends up circling aimlessly. The open-ended task, Buy Supply, shows that even without in-game guidance, CRADLE still manages to complete the task with its superior reasoning ability.

《荒野大镖客救赎2》。如图7所示，CRADLE在简单任务中表现出较高成功率，如跟随NPC或前往地面特定地点（例如，跟随Dutch和前往谷仓）。另一个跟随任务“跟随Javier”和搜索任务“搜索John”因通往雪山的崎岖蜿蜒路径及悬崖而较为危险。此外，GPT-4o在实时战斗和搜索任务中表现不佳，因其无法准确定位敌人或物体，也难以精准把握决策时机。即使配备了额外的检测工具，如Grounding DINO [35]，在需要夜间战斗的“保护Dutch”任务中成功率仍显著下降至20%。此外，室内任务如“搜索补给”和“搜索谷仓”也具有挑战性，因GPT-4o空间感知能力较差，难以定位目标物体，最终无目的地绕圈。开放式任务“购买补给”显示，即使没有游戏内指导，CRADLE依然凭借其卓越的推理能力完成了任务。

Stardew Valley. As shown in Table 1, we surprisingly find that GPT-4o struggles with accurately recognizing and locating objects near the player in this 2D game. This leads to difficulties for the agent to interact with objects or people, as it requires the player to stand precisely in front of them in the grid (e.g., when entering doors, using a pickaxe to break stones). It explains the inefficiency in the farming task though the agent manages to clear

《星露谷物语》。如表1所示，我们惊讶地发现GPT-4o在这款2D游戏中难以准确识别和定位玩家附近的物体。这导致代理与物体或人物交互时遇到困难，因为玩家需要精确站在格子前（例如，进入门、使用镐破石）。这解释了尽管代理能够清理农场任务，但效率不高的原因。

Table 1: CRADLE in Stardew Valley.

表1: CRADLE在《星露谷物语》中的表现。

Task	Results
Farm Clearup (Grids Num.)	14.8±5.0
Cultivation Shopping	4/5
	\$1/5\$
任务	结果
农场清理（网格数量）	14.8±5.0
耕作购物	4/5
	\$1/5\$

up most of the obstacles in front of the house within 100 steps (as shown in Figure 8a and 8b) and poor performance in the shopping task. On the other hand, relying on episodic summarization and task inference, CRADLE manages to obtain the parsnip by watering the seed for four days and harvesting.

在房屋前100步内的大部分障碍物（如图8a和8b所示）以及购物任务中的表现不佳。另一方面，依靠情节总结和任务推断，CRADLE成功通过浇水四天并收获，获得了欧防风（parsnip）。

Dealer's Life 2. Table 2 shows that CRADLE demonstrates robust performance and efficient profit-making on the Weekly Shop Management task, successfully finalizing 93.6% of potential transactions, with an average of 2 negotiation rounds per customer, and generally aiming for a profit rate of over 50% at the initial offer. It consistently generates profit across all runs, maintaining a total profit rate of +39.6%, peaking at +87.4% in a single run.

Dealer's Life 2。表2显示，CRADLE在每周商店管理任务中表现稳健且盈利高效，成功完成了93.6%的潜在交易，平均每位顾客进行2轮谈判，且通常在初始报价时目标利润率超过50%。它在所有运行中持续产生利润，总体利润率达到+39.6%，单次运行最高达到+87.4%。

Table 2: CRADLE in Dealer's Life 2.

表2：CRADLE在Dealer's Life 2中的表现。

Metrics	Results
Avg. Haggling Count	1.95±0.43
Turnover Rate (%)	93.6±6.9
Item Profit Rate (%)	37.8 ± 19.1
Total Profit Rate (%)	39.6 ± 27.3
指标	结果
平均讨价还价次数	1.95±0.43
周转率 (%)	93.6±6.9
单品利润率 (%)	37.8 ± 19.1
总利润率 (%)	39.6 ± 27.3



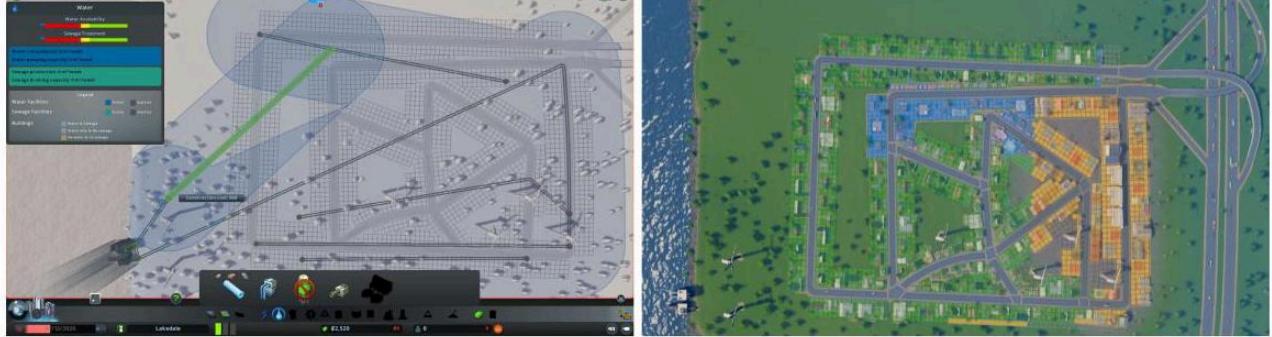
Figure 8: Three tasks in Stardew Valley. Each task is run with a maximum of 100 steps.

图8：星露谷物语中的三个任务。每个任务最多运行100步。

Figure 9: The task in Dealer's Life 2 of

100 steps. of running the shop for a week.

图9：Dealer's Life 2中经营商店一周的任务。



(a) CRADLE fails to connect all water pipes and cover all zones. Missing pipe is shown in green.

(b) The city built by CRADLE of 1100+ people after human assistance to get water pipes connected.

Figure 10: CRADLE's craft work in Cities: Skylines, shown in water view (left) and zoning view (right) with the same city. After fixing the error shown on the left, the city can finally be built into what is shown on the right. Each task is run with a maximum of 1000 steps or the budget is used up.

图10: CRADLE在《城市：天际线》中的建造工作，左图为水视图，右图为分区视图，均为同一城市。修正左图所示错误后，城市最终可建成右图所示状态。每个任务最多运行1000步或预算用尽。

Cities: Skylines. Table 3 shows that while CRADLE manages to build the roads in a closed loop to ensure smooth traffic flow, place multiple wind turbines to provide sufficient electricity supply and cover more than 90% of available area with residential, commercial and industrial zones, it fails to provide sufficient water supply reliably. As shown in Figure 10a, the most common failure case is that water pipes are not connected with each other, resulting in localized water shortages in the city, and preventing new residents from moving in. As shown in Figure 10b,

《城市：天际线》。表3显示，CRADLE能够构建闭环道路以确保交通流畅，布置多个风力涡轮机以提供充足电力，并用住宅、商业和工业区覆盖超过90%的可用区域，但未能可靠地提供足够的供水。如图10a所示，最常见的失败情况是水管未相互连接，导致城市局部缺水，阻碍新居民入住。如图10b所示，

Table 3: CRADLE in Cities: Skylines.

表3: 《城市: 天际线》中的CRADLE表现。

Task & Metrics	Results
Roads in Closed Loop	4/5
Sufficient Water Supply	1/5
Sufficient Electricity Supply	5/5
Zones Area $\geq 90\%$	4/5
Maximal Population	450±224
Maximal Population-w-HA	850±142
<hr/>	
任务与指标	结果
闭环道路	4/5
充足的供水	1/5
充足的供电	5/5
区域面积 $\geq 90\%$	4/5
最大人口	450±224
最大人口-含HA	850±142

with human assistance (-w-HA) to correct the mistakes within three unit operations (building or removing a road/facility/a place of zones is counted as one unit operation), the city built by CRADLE can eventually reach a population of more than one thousand. Table 3 also indicates that CRADLE nearly completes the city design, albeit with a few mistakes or omissions. With human assistance to fix these small issues, the final population can be doubled.

在人工协助 (-w-HA) 下纠正三个单元操作中的错误（建造或拆除一条道路/设施/区域地点计为一个单元操作），CRADLE构建的城市最终可以达到超过一千的人口规模。表3还显示，CRADLE几乎完成了城市设计，尽管存在一些错误或遗漏。通过人工协助修正这些小问题，最终人口数量可以翻倍。

Software Applications. Figure 11 shows CRADLE's success rates across tasks for all five applications. Multiple tasks remain challenging. Even with a well-known GUI, like Chrome and Outlook, GPT-4o still cannot recognize specific UI items to interact with and also struggles with visual context. For example, forgetting to press the Save button in an open dialog, or not distinguishing between a nearby enabled button vs. a distant and disabled one (e.g., when posting on Twitter). The phenomenon is more severe in the UI with non-standard layouts, like CapCut, Meitu, and Feishu. Lacking prior knowledge by GPT-4o leads to the failure of task inference and selecting the correct skills.

软件应用。图11展示了CRADLE在五个应用中各任务的成功率。多个任务仍然具有挑战性。即使是像Chrome和Outlook这样知名的图形用户界面 (GUI)，GPT-4o仍无法识别特定的UI元素进行交互，并且在视觉上下文理解上也存在困难。例如，忘记在打开的对话框中点击保存按钮，或无法区分附近的可用按钮与远处的禁用按钮（例如在Twitter发帖时）。这种现象在布局非标准的界面中更为严重，如CapCut、美图和飞书。GPT-4o缺乏先验知识导致任务推断失败和技能选择错误。

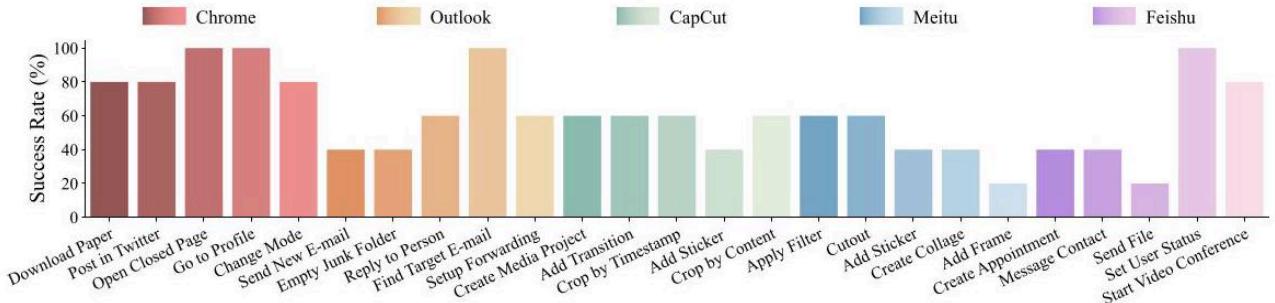


Figure 11: Success rates for tasks in software applications. Each task is run with a maximum of 30 steps.

图11：软件应用中各任务的成功率。每个任务最多执行30步。

OSWorld. Table 4 shows that CRADLE achieves the overall highest success rate in OSWorld, compared to the baselines without relying on any internal APIs to provide extra grounding labels, e.g., Set-of-Mark (SoM) [68]. The information gathering module improves grounding for more precise ac-  
OSWorld。表4显示，CRADLE在OSWorld中实现了总体最高的成功率，相较于不依赖任何内部API提供额外定位标签的基线方法，如Set-of-Mark (SoM) [68]。信息收集模块提升了定位的准确性，进而提高了精确的...

Table 4: Success rates (%) of different methods on OSWorld tasks.

表4：不同方法在OSWorld任务中的成功率（%）。

Method	Office (117)	OS (24)	Daily (78)	Workflow(101)	Professional (49)	All (369)
GPT-40	3.58	8.33	6.07	5.58	4.08	5.03
GPT-4o+SoM	3.58	20.83	3.99	3.60	2.04	4.59
Cradle	3.58	16.67	6.55	5.48	20.41	7.81
方法	办公 (117)	操作系统 (OS) (24)	日常 (78)	工作流程 (101)	专业的 (49)	全部 (369)
GPT-40	3.58	8.33	6.07	5.58	4.08	5.03
GPT-4o+SoM	3.58	20.83	3.99	3.60	2.04	4.59
摇篮	3.58	16.67	6.55	5.48	20.41	7.81

tion execution, increasing the performance. The self-reflection module greatly helps it to correctly predict infeasible tasks and subsequently fix mistakes, as exemplified in the professional domain results, where it achieves a 20.41% success rate, significantly surpassing the baselines.

执行任务，提高性能。自我反思模块极大地帮助其正确预测不可行任务并随后修正错误，如专业领域结果所示，其成功率达到20.41%，显著超越基线方法。

### 10.5 4.3 Ablation Study & Baselines Comparison

### 10.6 4.3 消融研究与基线比较

Since no existing methods are fully applicable to the GCC setting, we select several representative methods with necessary adaptions to make them applicable to GCC, labeling them as "like" in Table 6. Compared to CRADLE, React [72]-like method only has gather information, skill curation and action planning module, while Reflexion [53]-like method adds a self-reflection and episodic memory, compared to React-like. To show the necessity of multimodal input without access to APIs, we let GPT-40 describe the image and then feed the textual description to Voyager [57]-like as input. Additionally, experiments with GPT-40 and Claude 3 Opus [1] as backbone are conducted. Due to the limitation of requests per minute, other prompting methods like self-consistency [59] and TOT [71] are not considered.

由于现有方法均不完全适用于GCC设置，我们选择了几种具有代表性的方法并进行必要调整以适配GCC，在表6中标注为“like”。与CRADLE相比，React [72]-like方法仅包含信息收集、技能策划和行动规划模块，而Reflexion [53]-like方法在React-like基础上增加了自我反思和情节记忆。为展示无API访问时多模态输入的必要性，我们让GPT-4o描述图像，再将文本描述作为输入提供给Voyager [57]-like方法。此外，还进行了以GPT-4o和Claude 3 Opus [1]为骨干的实验。由于每分钟请求次数限制，未考虑自洽性 [59] 和TOT [71]等其他提示方法。

As seen in Table 6, all the baseline methods can only complete simple and straightforward tasks without complex targets and time delays. Compared to React-like method, Reflexion-like method has better performance in the task of Follow Micah and still fails to complete more complex tasks, emphasizing the importance of task inference and procedural memory. Voyager-like method that loses vision suffers to accomplish tasks and are the worst of all comparison methods. CRADLE with GPT-4o always has the best performance across all tasks. CRADLE with GPT-4o has the best performance, while Claude 3 Opus fails frequently due to unreliable OCR ability of the

guidance, leading to incorrect skill generation and failures of complex tasks.

如表6所示，所有基线方法仅能完成简单直接、无复杂目标和时间延迟的任务。与React-like方法相比，Reflexion-like方法在“跟随Micah”任务中表现更佳，但仍未能完成更复杂任务，凸显任务推断和程序记忆的重要性。失去视觉输入的Voyager-like方法难以完成任务，表现最差。搭载GPT-4o的CRADLE在所有任务中始终表现最佳。搭载GPT-4o的CRADLE表现最佳，而Claude 3 Opus因指导中的OCR能力不可靠，导致技能生成错误和复杂任务失败，频繁失败。

Figure 5 provides the detailed performance of each baseline method in the Cultivation task in Stardew Valley. Without task inference and episodic memory for summarization, even React-like and Reflexion-like methods sometimes managed to get the parsnip to sprout from the ground, they failed to successfully harvest it because GPT-4o failed to recognize the mature parsnip. Episodic summary can help CRADLE record the days of watering and know when the crop can be harvested. Voyager-like method struggles with getting out of the house and returning home due to the lack of visual input. Claude 3 Opus also has difficulties in localizing the position of the character and the crop. Moreover, it prefers moving characters much more frequently than GPT-4, resulting in the failure to position the character in front of the crop.

图5展示了各基线方法在《星露谷物语》耕作任务中的详细表现。缺乏任务推断和情节记忆总结，即使React-like和Reflexion-like方法有时能让欧防风从地里发芽，也未能成功收获，因为GPT-4o未能识别成熟的欧防风。情节总结帮助CRADLE记录浇水天数并判断作物成熟时间。缺乏视觉输入的Voyager-like方法难以走出房屋和返回家中。Claude 3 Opus在定位角色和作物位置上也存在困难，且比GPT-4更频繁地移动角色，导致无法将角色定位于作物前方。

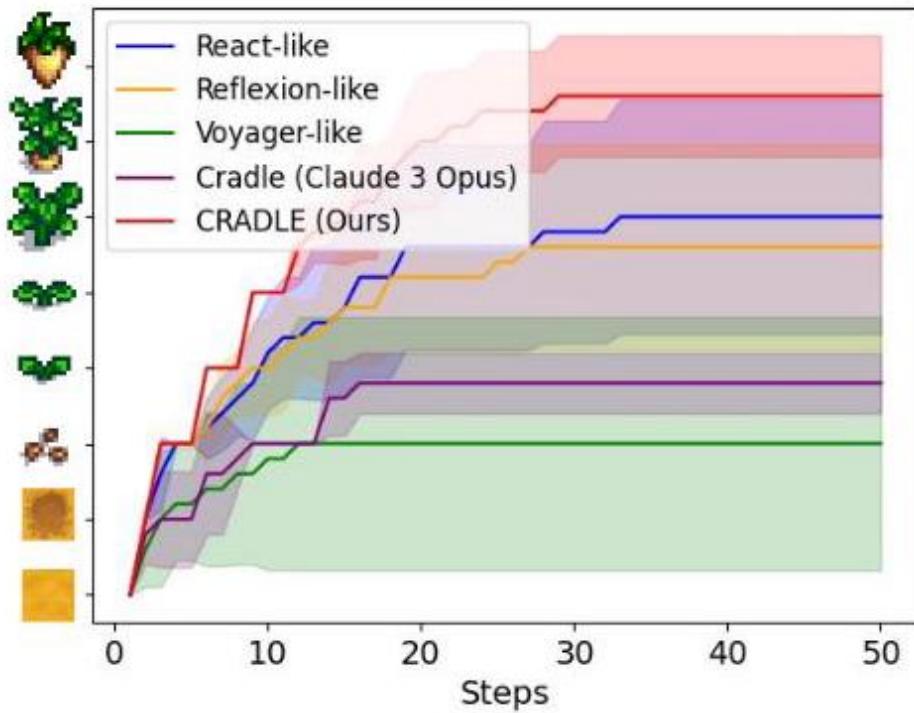


Table 5: Performance of each baseline method in task Cultivation. Only if the parsnip (shown on the top of the y-axis) is obtained will this run be counted as a success.

表5：《耕作》任务中各基线方法的表现。仅当获得欧防风（y轴顶部所示）时，该次运行才计为成功。

Table 6: Ablation results for RDR2 and Stardew Valley Cultivation tasks. Numbers before the brackets are interaction steps averaged over five trials. N/A indicates failure in all five runs. For RDR2, each task is run at most 500 steps and for Stardew Valley, each task is run for at most 100 steps.

表6：RDR2和《星露谷物语》耕作任务的消融结果。括号前数字为五次试验的平均交互步数。N/A表示五次均失败。RDR2每个任务最多运行500步，《星露谷物语》每个任务最多运行100步。

Method	Follow Dutch	Follow Micah	Hitch Horse	Protect Dutch	Search for Supplies	Cultivation
React [72]-like (GPT-40)	$\{15\} \pm 2 \left( \frac{5}{5} \right)$	$\{74\} \pm 0 \left( \frac{1}{5} \right)$	N/A	N/A	N/A	N/A
Reflextion [53]-like (GPT-40)	$\{19\} \pm 4 \left( \frac{5}{5} \right)$	$\{58\} \pm 14 \left( \frac{2}{5} \right)$	N/A	N/A	N/A	N/A
Voyager [57]-like (GPT-40)	$\{32\} \pm 12 \left( \frac{3}{5} \right)$	N/A	N/A	N/A	N/A	N/A
CRADLE (Claude 3 Opus)	$\{30\} \pm 7 \left( \frac{5}{5} \right)$	$\{52\} \pm 17 \left( \frac{4}{5} \right)$	N/A	N/A	N/A	N/A
CRADLE (GPT-40) (Ours)	$\{13\} \pm 3 \left( \frac{5}{5} \right)$	$\{33\} \pm 3 \left( \frac{5}{5} \right)$	$26 \pm 5$ (4/5)	$461 \pm 0$ (1/5)	$134 \pm 0$ (1/5)	$\{24\} \pm 4 \left( \frac{4}{5} \right)$

方法	跟随Dutch	跟随Micah	拴马	保护Dutch	寻找补给	耕作
类似React [72] (GPT-40)	$\{15\} \pm 2 \left( \frac{5}{5} \right)$	$\{74\} \pm 0 \left( \frac{1}{5} \right)$	不适用	不适用	不适用	不适用
类似Reflextion [53] (GPT-40)	$\{19\} \pm 4 \left( \frac{5}{5} \right)$	$\{58\} \pm 14 \left( \frac{2}{5} \right)$	不适用	不适用	不适用	不适用
类似Voyager [57] (GPT-40)	$\{32\} \pm 12 \left( \frac{3}{5} \right)$	不适用	不适用	不适用	不适用	不适用
CRADLE (Claude 3 Opus)	$\{30\} \pm 7 \left( \frac{5}{5} \right)$	$\{52\} \pm 17 \left( \frac{4}{5} \right)$	不适用	不适用	不适用	不适用
CRADLE (GPT-40) (我们的)	$\{13\} \pm 3 \left( \frac{5}{5} \right)$	$\{33\} \pm 3 \left( \frac{5}{5} \right)$	$26 \pm 5$ (4/5)	$461 \pm 0$ (1/5)	$134 \pm 0$ (1/5)	$\{24\} \pm 4 \left( \frac{4}{5} \right)$

## 11 5 Limitations and Future Work.

## 12 5 限制与未来工作。

Despite CRADLE's encouraging performance across games and software, several limitations remain. i) Due to the weaknesses of current LMM models, CRADLE struggles in recognizing out-of-distribution (OOD) icons and completing OOD tasks, such as games with non-realistic styles, i.e., Stardew Valley. As LMMs evolve, they can further improve CRADLE's performance. ii) Audio, as an important modality, often plays an important role in games and software; however, it has not been considered in this work. The future work will be enabling CRADLE to process the audio and graphical input simultaneously. iii) Most CRADLE's modules need to call LMM explicitly to process the input for best performance, resulting in frequent interactions with LMM and potentially high costs and long delays. The six modules represent a problem-solving mindset; as LMM capabilities improve, some or even all of these modules may be combined into a single request. iv) In this work, we mainly focus on enabling foundation agents to interact with various software in a unified manner without taking training into consideration. As SIMA [48] has already shown promising results in a similar setting with the trained agents, we will let CRADLE autonomously explore and improve over environments through RL [55] or collect expert demonstrations for supervised learning [48]. v) Though CRADLE is generally applicable to any computer task, only a few with limited tasks are investigated in this work. We will extend it to a wider range of targets, go deeper into the complex games, and make it easier for users to adapt on their own.

尽管CRADLE在多款游戏和软件中表现令人鼓舞，但仍存在若干限制。i) 由于当前大规模多模态模型（LMM）存在不足，CRADLE在识别分布外（OOD）图标和完成OOD任务（如非写实风格游戏，例如《星露谷物语》（Stardew Valley））方面表现较弱。随着LMM的发展，CRADLE的性能有望进一步提升。ii) 音频作为重要模态，在游戏和软件中常扮演关键角色；然而本工作未涉及音频处理。未来工作将使CRADLE能够同时处理音频和图形输入。iii)

CRADLE的大多数模块需显式调用LMM以实现最佳性能，导致频繁与LMM交互，可能带来高成本和较长延迟。这六个模块体现了解决问题的思路；随着LMM能力提升，部分甚至全部模块可能合并为单一请求。iv) 本文主要关注使基础代理以统一方式与各种软件交互，未考虑训练。鉴于SIMA [48]在类似设置下训练代理已展现出良好效果，我们计划让CRADLE通过强化学习（RL）[55]自主探索和改进环境，或收集专家示范进行监督学习[48]。v) 虽然CRADLE普遍适用于任何计算机任务，但本工作仅研究了少数有限任务。未来将扩展至更广泛目标，深入复杂游戏，并简化用户自适应过程。

CRADLE holds great potential to improve effective general computer task completion and boost research and deployment of foundation agents. However, there is also a risk of unintended or unsuitable usage, including developing game cheats, incorrect operations of software with harmful failures, or other negative agent behavior. Therefore, additional regulations or safeguards are required for secure and responsible deployments across digital and physical environments.

CRADLE具备显著潜力提升通用计算机任务完成效率，推动基础代理的研究与应用。然而，也存在误用或不当使用风险，包括开发游戏作弊、软件误操作导致有害故障或其他负面代理行为。因此，需制定额外规章或安全措施，确保在数字及物理环境中的安全与负责任部署。

## 13 6 Conclusion

## 14 6 结论

In this work, we introduce GCC, a general and challenging setting with a unified and standard interface for control of diverse video games and other software (via screenshots, and keyboard and mouse operations), paving the way towards general foundation agents across all digital world tasks. To properly address the challenges GCC presents, we propose a novel open-source framework, CRADLE, which exhibits strong performance in reasoning and performing actions to accomplish real missions or tasks in a set of complex video games and common software applications. To the best of our knowledge, CRADLE is the first framework that enables foundation agents to succeed in such a diverse set of environments without relying on any built-in APIs. The success of CRADLE greatly extends the reach of foundation agents and demonstrates the feasibility of converting any software, especially complex games, into benchmarks to evaluate agents' general intelligence and facilitate further data collection for self-improvement. Although CRADLE still faces difficulties in certain tasks, it serves as a pioneering work to develop more powerful LMM-based agents across computer control tasks, combining both further framework enhancements and new advances in LMMs.

本文提出了GCC，一种通用且具挑战性的环境，提供统一标准接口以控制多样视频游戏及其他软件（通过截图、键盘和鼠标操作），为实现覆盖所有数字世界任务的通用基础代理奠定基础。为应对GCC带来的挑战，我们提出了新颖的开源框架CRADLE，在推理和执行动作以完成复杂视频游戏及常用软件中的真实任务方面表现出色。据我们所知，CRADLE是首个无需依赖任何内置API即可使基础代理在如此多样环境中成功的框架。CRADLE的成功极大拓展了基础代理的应用范围，展示了将任何软件，尤其是复杂游戏，转化为评估代理通用智能的基准并促进自我提升数据收集的可行性。尽管CRADLE在某些任务上仍面临挑战，但其作为开创性工作，为基于LMM的更强大计算机控制代理的发展提供了基础，结合框架改进与LMM新进展。

## 15 7 Acknowledgments and Disclosure of Funding

## 16 7 致谢与资金披露

We thank Ye Wang and Jiangxing Wang for their time and effort in helping us test the framework. Xinrun Xu is advised by Dr. Zhiming Ding from the Institute of Software, Chinese Academy of Sciences, and is supported by the National Key R&D Program of China (No. 2022YFF0503900).

感谢王烨和王江星在测试框架过程中付出的时间和努力。徐新润由中国科学院软件研究所丁志明博士指导，获得国家重点研发计划（编号2022YFF0503900）资助。

## **17 8 Team Members and Contributions**

### **18 8 团队成员与贡献**

#### **18.1 8.1 Roles**

##### **18.2 8.1 角色分工**

Program Leads: Zongqing Lu, Shuicheng Yan, and Bo An

项目负责人：卢宗庆、严水成、安博

Team Lead: Weihao Tan

团队负责人：谭伟豪

Framework Co-Leads: Börje F. Karlsson and Weihao Tan

框架联合负责人：Börje F. Karlsson、谭伟豪

General Advisors: Xinrun Wang and Börje F. Karlsson

总体顾问：王新润、Börje F. Karlsson

Core Contributors: Weihao Tan, Wentao Zhang, Xinrun Xu, Haochong Xia, Ziluo Ding, Boyu Li, Bohan Zhou, Junpeng Yue, Jiechuan Jiang, Yewen Li, Ruyi An, Molei Qin, Chuqiao Zong, Longtao Zheng, Yujie Wu, Xiaoqiang Chai, Xinrun Wang, and Börje F. Karlsson

核心贡献者：谭伟豪、张文涛、徐新润、夏浩冲、丁子洛、李博宇、周博涵、岳俊鹏、姜杰川、李业文、安如意、秦墨磊、宗楚乔、郑龙涛、吴宇杰、柴晓强、王新润、Börje F. Karlsson

#### **18.3 8.2 Detailed Contributions**

##### **18.4 8.2 详细贡献**

Framework Design. Weihao Tan, Börje F. Karlsson, Wentao Zhang, Ziluo Ding, and Xinrun Wang.

框架设计：谭伟豪、Börje F. Karlsson、张文涛、丁子洛、王新润。

RDR2. Implementation, experiments, and analysis by Weihao Tan, Börje F. Karlsson, Wentao Zhang, Ziluo Ding, Boyu Li, Bohan Zhou, Junpeng Yue, Haochong Xia, Jiechuan Jiang, Molei Qin, Longtao Zheng, Xinrun Xu, Yifei Bi, Pengjie Gu, and Yewen Li. Xinrun Wang provided insightful suggestions.

RDR2。由谭伟豪、博耶·F·卡尔松、张文涛、丁子洛、李博宇、周博涵、岳俊鹏、夏浩冲、姜杰川、秦墨雷、郑龙涛、徐新润、毕一飞、顾鹏杰和李业文负责实现、实验和分析。王新润提供了有见地的建议。

Stardew Valley. Implementation, experiments, and analysis by Weihao Tan, Wentao Zhang, Chuqiao Zong, Yujie Wu, Xiaoqiang Chai, Haochong Xia, Yewen Li, Ruyi An, and Molei Qin. Xinrun Wang, Long Tian, Chaojie Wang, and Börje F. Karlsson provided insightful suggestions.

星露谷物语。由谭伟豪、张文涛、宗楚乔、吴宇杰、柴晓强、夏浩冲、李业文、安如意和秦墨雷负责实现、实验和分析。王新润、田龙、王超杰和博耶·F·卡尔松提供了有见地的建议。

Cities: Skylines. Implementation, experiments, and analysis by Weihao Tan, Wentao Zhang, Haochong Xia, and Ceyao Zhang. Xinrun Wang provided insightful suggestions.

城市：天际线。由谭伟豪、张文涛、夏浩冲和张策尧负责实现、实验和分析。王新润提供了有见地的建议。

Dealer's Life 2. Implementation, experiments, and analysis by Yewen Li, Ruyi An. Weihao Tan, Wentao Zhang, and Xinrun Wang provided insightful suggestions.

经销商生活2。由李业文、安如意负责实现、实验和分析。谭伟豪、张文涛和王新润提供了有见地的建议。

Software Applications (including Chrome, Outlook, CapCut, Meitu, and Feishu). Implementation, experiments, and analysis by Xinrun Xu, Börje F. Karlsson, and Xiyun Li. Weihao Tan and Xinrun Wang provided insightful suggestions.

软件应用（包括Chrome、Outlook、CapCut、美图和飞书）。由徐新润、博耶·F·卡尔松和李希云负责实现、实验和分析。谭伟豪和王新润提供了有见地的建议。

OSWorld. Implementation, experiments, and analysis by Haochong Xia, Tianbao Xie, Pengjie Gu. Weihao Tan, Xinrun Xu, Xinrun Wang, and Börje F. Karlsson provided insightful suggestions.

OSWorld。由夏浩冲、谢天宝、顾鹏杰负责实现、实验和分析。谭伟豪、徐新润、王新润和博耶·F·卡尔松提供了有见地的建议。

Paper Writing. Weihao Tan, Xinrun Wang, Börje F. Karlsson, Wentao Zhang, Xinrun Xu, Yewen Li, Ruyi An, and Chuqiao Zong. Haochong Xia, Molei Qin, and Ceyao Zhang further contributed with writers in appendix organization.

论文写作。谭伟豪、王新润、博耶·F·卡尔松、张文涛、徐新润、李业文、安如意和宗楚乔。夏浩冲、秦墨雷和张策尧在附录组织方面进一步贡献。

Organization. Zongqing Lu, Shuicheng Yan, and Bo An provided directional research advice and organizational support.

组织。卢宗庆、闫水成和安博提供了方向性研究建议和组织支持。

## 19 References

## 20 参考文献

[1] Anthropic. The claude 3 model family: Opus, sonnet, haiku, 2024.

[1] Anthropic。Claude 3模型系列：Opus、Sonnet、Haiku，2024年。

[2] Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (VPT): Learning to act by watching unlabeled online videos. Advances in Neural Information Processing Systems, 35:24639-24654, 2022.

[2] Bowen Baker, Ilge Akkaya、Peter Zhokov、Joost Huizinga、Jie Tang、Adrien Ecoffet、Brandon Houghton、Raul Sampedro和Jeff Clune。视频预训练（VPT）：通过观看无标签在线视频学习行动。神经信息处理系统进展，35：24639-24654，2022年。

[3] Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. Science, 378(6624):1067-1074, 2022.

[3] Anton Bakhtin、Noam Brown、Emily Dinan、Gabriele Farina、Colin Flaherty、Daniel Fried、Andrew Goff、Jonathan Gray、Hengyuan Hu等。通过结合语言模型与战略推理实现《外交》游戏的人类水平玩法。科学，378(6624)：1067-1074，2022年。

[4] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The Arcade learning environment: An evaluation platform for general agents. Journal of Artificial Intelligence Research, 47:253-279, 2013.

[4] Marc G Bellemare、Yavar Naddaf、Joel Veness和Michael Bowling。街机学习环境：通用智能体的评估平台。人工智能研究杂志，47：253-279，2013年。

[5] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. arXiv preprint arXiv:1912.06680, 2019.

[5] Christopher Berner、Greg Brockman、Brooke Chan、Vicki Cheung、Przemysław Dębiak、Christy Dennison、

David Farhi、Quirin Fischer、Shariq Hashme、Chris Hesse等。基于大规模深度强化学习的Dota 2。arXiv预印本 arXiv:1912.06680，2019年。

[6] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. arXiv preprint arXiv:2307.15818, 2023.

[6] Anthony Brohan、Noah Brown、Justice Carbajal、Yevgen Chebotar、Xi Chen、Krzysztof Choromanski、Tianli Ding、Danny Driess、Avinava Dubey、Chelsea Finn等。RT-2：视觉-语言-动作模型将网络知识转移到机器人控制。arXiv预印本 arXiv:2307.15818，2023年。

[7] Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, et al. Do as I can, not as I say: Grounding language in robotic affordances. In Conference on Robot Learning, pages 287-318. PMLR, 2023.

[7] Anthony Brohan、Yevgen Chebotar、Chelsea Finn、Karol Hausman、Alexander Herzog、Daniel Ho、Julian Ibarz、Alex Irpan、Eric Jang、Ryan Julian等。言行合一：将语言基础植入机器人可操作性。机器人学习会议论文集，页287-318。PMLR，2023年。

[8] Roberto Brunelli. Template matching techniques in computer vision: theory and practice. John Wiley & Sons, 2009.

[8] Roberto Brunelli. 计算机视觉中的模板匹配技术：理论与实践。John Wiley & Sons, 2009。

[9] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. Advances in neural information processing systems, 32, 2019.

[9] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, 和 Anca Dragan. 关于学习人类行为以促进人机协作的效用。神经信息处理系统进展, 32, 2019。

[10] Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. SeeClick: Harnessing GUI grounding for advanced visual GUI agents. arXiv preprint arXiv:2401.10935, 2024.

[10] 程坤志, 孙秋实, 褚有刚, 徐方志, 李彦涛, 张建兵, 和 吴志勇. SeeClick：利用GUI定位实现高级视觉GUI代理。arXiv预印本 arXiv:2401.10935, 2024。

[11] Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. BabyAI: First steps towards grounded language learning with a human in the loop. In International Conference on Learning Representations, 2019.

[11] Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, 和 Yoshua Bengio. BabyAI：迈向带有人类参与的基础语言学习的第一步。国际学习表征会议论文集, 2019。

[12] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. Advances in Neural Information Processing Systems, 35:5982-5994, 2022.

[12] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, 和 Roozbeh Mottaghi. Procthor：利用程序生成的大规模具身人工智能。神经信息处理系统进展, 35:5982-5994, 2022。

[13] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. arXiv preprint arXiv:2306.06070, 2023.

[13] 邓翔, 顾宇, 郑博远, 陈世杰, Samuel Stevens, 王博实, 孙焕, 和 苏宇. Mind2Web：迈向通用网络代理。arXiv预印本 arXiv:2306.06070, 2023。

- [14] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378, 2023.
- [14] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, 等. Palm-e: 一个具身多模态语言模型。arXiv预印本 arXiv:2303.03378, 2023。
- [15] Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Nicolaus Foerster, and Shimon Whiteson. SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning. In Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2023.
- [15] Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Nicolaus Foerster, 和 Shimon Whiteson. SMACv2: 改进的合作多智能体强化学习基准。第三十七届神经信息处理系统会议数据集与基准赛道, 2023。
- [16] Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. Advances in Neural Information Processing Systems, 35:18343-18362, 2022.
- [16] 范林曦, 王冠志, 蒋云帆, Ajay Mandlekar, 杨云聪, 朱浩毅, Andrew Tang, 黄德安, 朱宇珂, 和 Anima Anandkumar. Minedojo: 构建具备互联网规模知识的开放式具身代理。神经信息处理系统进展, 35:18343-18362, 2022。
- [17] Hiroki Furuta, Ofir Nachum, Kuang-Huei Lee, Yutaka Matsuo, Shixiang Shane Gu, and Izzeddin Gur. Multimodal web navigation with instruction-finetuned foundation models. arXiv preprint arXiv:2305.11854, 2023.
- [17] 古田浩树, Ofir Nachum, Kuang-Huei Lee, 松尾丰, Shixiang Shane Gu, 和 Izzeddin Gur. 基于指令微调基础模型的多模态网页导航。arXiv预印本 arXiv:2305.11854, 2023。
- [18] Difei Gao, Lei Ji, Zechen Bai, Mingyu Ouyang, Peiran Li, Dongxing Mao, Qinchen Wu, Weichen Zhang, Peiyi Wang, Xiangwu Guo, et al. ASSISTGUI: Task-oriented desktop graphical user interface automation. arXiv preprint arXiv:2312.13108, 2023.
- [18] 高迪飞, 纪磊, 白泽辰, 欧阳明宇, 李沛然, 毛东兴, 吴秦辰, 张伟晨, 王培怡, 郭祥武, 等. ASSISTGUI: 面向任务的桌面图形用户界面自动化。arXiv预印本 arXiv:2312.13108, 2023。
- [19] Xiaofeng Gao, Ran Gong, Tianmin Shu, Xu Xie, Shu Wang, and Song-Chun Zhu. Vrkitchen: an interactive 3d virtual environment for task-oriented learning. arXiv preprint arXiv:1903.05757, 2019.
- [19] 高晓峰, 龚然, 舒天民, 谢旭, 王舒, 和 朱松纯. Vrkitchen: 一个用于任务导向学习的交互式3D虚拟环境。arXiv预印本 arXiv:1903.05757, 2019。
- [20] Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. A real-world webagent with planning, long context understanding, and program synthesis. arXiv preprint arXiv:2307.12856, 2023.
- [20] Izzeddin Gur, 古田浩树, Austin Huang, Mustafa Safdari, 松尾丰, Douglas Eck, 和 Aleksandra Faust. 一个具备规划、长上下文理解和程序合成能力的真实世界网页代理。arXiv预印本 arXiv:2307.12856, 2023。
- [21] William H Guss, Brandon Houghton, Nicholay Topin, Phillip Wang, Cayden Codel, Manuela Veloso, and Ruslan Salakhutdinov. Minerl: A large-scale dataset of Minecraft demonstrations. arXiv preprint arXiv:1907.13440, 2019.
- [21] William H Guss, Brandon Houghton, Nicholay Topin, Phillip Wang, Cayden Codel, Manuela Veloso, 和 Ruslan Salakhutdinov. Minerl: 一个大规模Minecraft演示数据集。arXiv预印本 arXiv:1907.13440, 2019。

- [22] Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an end-to-end web agent with large multimodal models. arXiv preprint arXiv:2401.13919, 2024.
- [22] 何洪亮, 姚文林, 马凯欣, 余文浩, 戴勇, 张洪明, 兰振中, 余东。WebVoyager: 基于大型多模态模型构建端到端网页代理。arXiv预印本 arXiv:2401.13919, 2024年。
- [23] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. CogAgent: A visual language model for GUI agents. arXiv preprint arXiv:2312.08914, 2023.
- [23] 洪文怡, 王伟涵, 吕庆松, 徐家正, 余文萌, 季俊辉, 王岩, 王子涵, 董宇霄, 丁明, 等。CogAgent: 面向GUI代理的视觉语言模型。arXiv预印本 arXiv:2312.08914, 2023年。
- [24] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. arXiv preprint arXiv:2207.05608, 2022.
- [24] 黄文龙, 夏飞, 肖特德, 陈哈里斯, 梁杰奇, 弗洛伦斯·皮特, 曾安迪, 汤普森乔纳森, 莫尔达奇伊戈尔, 切博塔耶夫叶夫根, 等。内心独白: 通过语言模型规划实现具身推理。arXiv预印本 arXiv:2207.05608, 2022年。
- [25] Max Jaderberg, Wojciech M Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castaneda, Charles Beattie, Neil C Rabinowitz, Ari S Morcos, Avraham Ruderman, et al. Human-level performance in 3D multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859-865, 2019.
- [25] 马克斯·贾德伯格, 沃伊切赫·M·查尔内茨基, 伊恩·邓宁, 卢克·马里斯, 盖伊·莱弗, 安东尼奥·加西亚·卡斯塔涅达, 查尔斯·比蒂, 尼尔·C·拉宾诺维茨, 阿里·S·莫科斯, 阿夫拉罕·鲁德曼, 等。基于群体强化学习的3D多人游戏中达到人类水平表现。科学, 364(6443):859-865, 2019年。
- [26] Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The Malmo platform for artificial intelligence experimentation. In *Ijcai*, pages 4246-4247, 2016.
- [26] 马修·约翰逊, 卡佳·霍夫曼, 蒂姆·哈顿, 大卫·比格内尔。Malmo平台: 人工智能实验平台。发表于国际人工智能联合会议 (IJCAI) , 第4246-4247页, 2016年。
- [27] Raghav Kapoor, Yash Parag Butala, Melisa Russak, Jing Yu Koh, Kiran Kamble, Waseem Al-shikh, and Ruslan Salakhutdinov. OmniACT: A dataset and benchmark for enabling multimodal generalist autonomous agents for desktop and web, 2024.
- [27] 拉加夫·卡普尔, 亚什·帕拉格·布塔拉, 梅丽莎·鲁萨克, 许静瑜, 基兰·坎布尔, 瓦西姆·阿尔-希克, 鲁斯兰·萨拉胡丁诺夫。OmniACT: 支持桌面和网页多模态通用自主代理的数据集与基准, 2024年。
- [28] Christian Kauten. Super Mario Bros for OpenAI Gym. GitHub, 2018.
- [28] 克里斯蒂安·考滕。OpenAI Gym的超级马里奥兄弟。GitHub, 2018年。
- [29] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015-4026, 2023.
- [29] 亚历山大·基里洛夫, 埃里克·明顿, 尼基拉·拉维, 毛汉子, 克洛伊·罗兰, 劳拉·古斯塔夫森, 肖特特, 斯宾塞·怀特黑德, 亚历山大·C·伯格, 罗万·洛, 等。Segment anything (任意分割)。发表于IEEE/CVF国际计算机视觉会议, 4015-4026页, 2023年。
- [30] Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. arXiv preprint arXiv:2401.13649, 2024.
- [30] 许静瑜, 罗伯特·洛, 劳伦斯·姜, 维克拉姆·杜弗, 林明聪, 黄柏宇, 格雷厄姆·纽比格, 周书妍, 鲁斯兰·萨拉胡丁诺夫, 丹尼尔·弗里德。VisualWebArena: 评估多模态代理在真实视觉网页任务中的表现。arXiv预印本 arXiv:2401.13649, 2024年。

- [31] Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zajac, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, et al. Google research football: A novel reinforcement learning environment. In Proceedings of the AAAI conference on artificial intelligence, pages 4501-4510, 2020.
- [31] 卡罗尔·库拉赫, 安东·赖丘克, 皮奥特·斯坦奇克, 米哈乌·扎亚茨, 奥利维耶·巴谢姆, 拉塞·埃斯佩霍尔特, 卡洛斯·里克尔梅, 达米安·文森特, 马尔钦·米哈尔斯基, 奥利维耶·布斯凯, 等。谷歌研究足球: 一种新颖的强化学习环境。发表于AAAI人工智能会议, 4501-4510页, 2020年。
- [32] Joel Z Leibo, Edgar A Duéñez-Guzman, Alexander Vezhnevets, John P Agapiou, Peter Sunehag, Raphael Koster, Jayd Matyas, Charlie Beattie, Igor Mordatch, and Thore Graepel. Scalable evaluation of multi-agent reinforcement learning with melting pot. In International conference on machine learning, pages 6187-6199. PMLR, 2021.
- [32] 乔尔·Z·莱博, 埃德加·A·杜涅兹-古兹曼, 亚历山大·韦兹涅维茨, 约翰·P·阿加皮乌, 彼得·苏内哈格, 拉斐尔·科斯特, 杰伊德·马蒂亚斯, 查理·比蒂, 伊戈尔·莫尔达奇, 托雷·格雷佩尔。使用Melting Pot进行多智能体强化学习的可扩展评估。发表于国际机器学习会议, 6187-6199页, PMLR, 2021年。
- [33] Chengshu Li, Fei Xia, Roberto Martín-Martín, Michael Lingelbach, Sanjana Srivastava, Bokui Shen, Kent Vainio, Cem Gokmen, Gokul Dharan, Tanish Jain, et al. igibson 2.0: Object-centric simulation for robot learning of everyday household tasks. arXiv preprint arXiv:2108.03272, 2021.
- [33] 李成书, 夏飞, 罗伯托·马丁-马丁, 迈克尔·林格尔巴赫, 桑贾纳·斯里瓦斯塔瓦, 申博奎, 肯特·瓦伊尼奥, 杰姆·戈克门, 戈库尔·达兰, 塔尼什·贾因, 等。iGibson 2.0: 面向机器人学习日常家务任务的以物体为中心的仿真环境。arXiv预印本 arXiv:2108.03272, 2021年。
- [34] Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tianlin Shi, and Percy Liang. Reinforcement learning on web interfaces using workflow-guided exploration. In International Conference on Learning Representations (ICLR), 2018.
- [34] 刘哲然, 郭凯文, 帕努蓬·帕苏帕特, 石天霖, 梁帕西。基于工作流引导探索的网页界面强化学习。发表于国际学习表征会议 (ICLR) , 2018年。
- [35] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding Dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499, 2023.
- [35] 刘世龙, 曾昭阳, 任天河, 李峰, 张浩, 杨杰, 李春元, 杨建伟, 苏航, 朱军, 等. Grounding Dino: 将Dino与基于定位的预训练结合用于开放集目标检测. arXiv预印本 arXiv:2303.05499, 2023.
- [36] Weiyu Ma, Qirui Mi, Xue Yan, Yuqiao Wu, Runji Lin, Haifeng Zhang, and Jun Wang. Large language models play StarCraft II: Benchmarks and a chain of summarization approach. arXiv preprint arXiv:2312.11865, 2023.
- [36] 马伟宇, 米启睿, 颜雪, 吴玉桥, 林润吉, 张海峰, 王军. 大型语言模型玩星际争霸II: 基准测试与链式摘要方法. arXiv预印本 arXiv:2312.11865, 2023.
- [37] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.
- [37] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, 赵一力, Erik Wijmans, Bhavana Jain, Julian Straub, 刘佳, Vladlen Koltun, Jitendra Malik, Devi Parikh, 和 Dhruv Batra. Habitat: 一个用于具身人工智能研究的平台. 载于IEEE/CVF国际计算机视觉会议(ICCV)论文集, 2019.
- [38] Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. arXiv preprint arXiv:2311.12983, 2023.
- [38] Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, 和 Thomas Scialom. GAIA: 通用人工智能助手的基准测试. arXiv预印本 arXiv:2311.12983, 2023.

- [39] Meredith Ringel Morris, Jascha Sohl-dickstein, Noah Fiedel, Tris Warkentin, Allan Dafoe, Aleksandra Faust, Clement Farabet, and Shane Legg. Levels of AGI: Operationalizing progress on the path to AGI. arXiv preprint arXiv:2311.02462, 2023.
- [39] Meredith Ringel Morris, Jascha Sohl-dickstein, Noah Fiedel, Tris Warkentin, Allan Dafoe, Aleksandra Faust, Clement Farabet, 和 Shane Legg. AGI的层级：实现通用人工智能进展的操作化路径. arXiv预印本 arXiv:2311.02462, 2023.
- [40] Runliang Niu, Jindong Li, Shiqi Wang, Yali Fu, Xiyu Hu, Xueyuan Leng, He Kong, Yi Chang, and Qi Wang. ScreenAgent: A vision language model-driven computer control agent. arXiv preprint arXiv:2402.07945, 2024.
- [40] 牛润良, 李金东, 王世奇, 傅雅丽, 胡希宇, 冷学元, 孔贺, 常毅, 和 王琦. ScreenAgent: 一个视觉语言模型驱动的计算机控制代理. arXiv预印本 arXiv:2402.07945, 2024.
- [41] OpenAI. Universe, 2016.
- [41] OpenAI. Universe, 2016.
- [42] OpenAI. New and improved embedding model, 2022.
- [42] OpenAI. 新版改进的嵌入模型, 2022.
- [43] OpenAI. Hello gpt-4o, 2024.
- [43] OpenAI. 你好 gpt-4o, 2024.
- [44] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, pages 1-22, 2023.
- [44] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, 和 Michael S Bernstein. 生成代理：人类行为的交互式模拟. 载于第36届ACM用户界面软件与技术年会论文集, 页码1-22, 2023.
- [45] Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. Virtualhome: Simulating household activities via programs. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 8494-8502, 2018.
- [45] Xavier Puig, Kevin Ra, Marko Boben, 李佳曼, 王廷武, Sanja Fidler, 和 Antonio Torralba. Virtualhome: 通过程序模拟家庭活动. 载于IEEE计算机视觉与模式识别会议论文集, 页码8494-8502, 2018.
- [46] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. arXiv preprint arXiv:2310.13724, 2023.
- [46] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, 杨宗彦, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, Min So Yeon, 等. Habitat 3.0: 人类、虚拟形象与机器人共存的环境. arXiv 预印本 arXiv:2310.13724, 2023.
- [47] Siyuan Qi, Shuo Chen, Yixin Li, Xiangyu Kong, Junqi Wang, Bangcheng Yang, Pring Wong, Yifan Zhong, Xiaoyuan Zhang, Zhaowei Zhang, et al. CivRealm: A learning and reasoning odyssey in Civilization for decision-making agents. In ICLR, 2024.
- [47] 齐思远, 陈硕, 李业新, 孔祥宇, 王俊奇, 杨邦成, Pring Wong, 钟一凡, 张晓媛, 张兆伟, 等. CivRealm: 文明游戏中面向决策代理的学习与推理探索. 载于ICLR, 2024.
- [48] Maria Abi Raad, Arun Ahuja, Catarina Barros, Frederic Besse, Andrew Bolt, Adrian Bolton, Bethanie Brownfield, Gavin Buttimore, Max Cant, Sarah Chakera, et al. Scaling instructable agents across many simulated worlds. arXiv preprint arXiv:2404.10179, 2024.
- [48] Maria Abi Raad, Arun Ahuja, Catarina Barros, Frederic Besse, Andrew Bolt, Adrian Bolton, Bethanie Brownfield, Gavin Buttimore, Max Cant, Sarah Chakera, 等. 跨多个模拟世界扩展可指导代理. arXiv预印本 arXiv:2404.10179, 2024.

- [49] Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. Android in the wild: A large-scale dataset for Android device control. arXiv preprint arXiv:2307.10088, 2023.
- [49] Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, 和 Timothy Lillicrap. Android in the wild: 一个用于Android设备控制的大规模数据集. arXiv预印本 arXiv:2307.10088, 2023.
- [50] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nan-tas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. The Starcraft multi-agent challenge. arXiv preprint arXiv:1902.04043, 2019.
- [50] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, 和 Shimon Whiteson. 星际争霸多智能体挑战. arXiv预印本 arXiv:1902.04043, 2019.
- [51] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne Tchapmi, et al. igibson 1.0: a simulation environment for interactive tasks in large realistic scenes. In 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 7520-7527. IEEE, 2021.
- [51] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne Tchapmi, 等. igibson 1.0: 用于大规模真实场景交互任务的仿真环境. 载于2021年IEEE/RSJ国际智能机器人与系统会议 (IROS) , 页7520-7527. IEEE, 2021.
- [52] Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, and Percy Liang. World of bits: An open-domain platform for web-based agents. In International Conference on Machine Learning, pages 3135-3144. PMLR, 2017.
- [52] Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, 和 Percy Liang. World of bits: 基于网络代理的开放域平台. 载于国际机器学习会议, 页3135-3144. PMLR, 2017.
- [53] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In Thirty-seventh Conference on Neural Information Processing Systems, 2023.
- [53] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, 和 Shunyu Yao. Reflexion: 具备语言强化学习的语言代理. 载于第三十七届神经信息处理系统会议, 2023.
- [54] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In Advances in Neural Information Processing Systems (NeurIPS), 2021.
- [54] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, 和 Dhruv Batra. Habitat 2.0: 训练家庭助理重新布置其生活环境. 载于神经信息处理系统进展 (NeurIPS) , 2021.
- [55] Weihao Tan, Wentao Zhang, Shanqi Liu, Longtao Zheng, Xinrun Wang, and Bo An. True knowledge comes from practice: Aligning large language models with embodied environments via reinforcement learning. In ICLR, 2023.
- [55] Weihao Tan, Wentao Zhang, Shanqi Liu, Longtao Zheng, Xinrun Wang, 和 Bo An. 真正的知识来自实践: 通过强化学习使大型语言模型与具身环境对齐. 载于ICLR, 2023.
- [56] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojcicek M Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, et al. AlphaStar: Mastering the real-time strategy game Starcraft II. DeepMind blog, 2:20, 2019.
- [56] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M Czarnecki,

Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, 等. AlphaStar: 掌握实时战略游戏星际争霸II. DeepMind博客, 2:20, 2019.

[57] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*, 2024.

[57] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, 和 Anima Anandkumar. Voyager: 具备大型语言模型的开放式具身代理. 机器学习研究汇刊, 2024.

[58] Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-Agent: Autonomous multi-modal mobile device agent with visual perception. arXiv preprint arXiv:2401.16158, 2024.

[58] Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, 和 Jitao Sang. Mobile-Agent: 具备视觉感知的自主多模态移动设备代理. arXiv预印本 arXiv:2401.16158, 2024.

[59] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:2203.11171, 2022.

[59] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, 和 Denny Zhou. 自治性提升语言模型中的链式思维推理. arXiv预印本 arXiv:2203.11171, 2022.

[60] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, and Yitao Liang. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. arXiv preprint arXiv:2311.05997, 2023.

[60] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, 和 Yitao Liang. Jarvis-1: 具备记忆增强多模态语言模型的开放世界多任务代理. arXiv预印本 arXiv:2311.05997, 2023.

[61] Zihao Wang, Shaofei Cai, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. In ICML, 2023.

[61] Zihao Wang, Shaofei Cai, Anji Liu, Xiaojian Ma, 和 Yitao Liang. 描述、解释、规划与选择：利用大型语言模型的交互式规划实现开放世界多任务代理. 载于ICML, 2023.

[62] Sarah A. Wu, Rose E. Wang, James A. Evans, Joshua B. Tenenbaum, David C. Parkes, and Max Kleiman-Weiner. Too many cooks: Coordinating multi-agent collaboration through inverse planning. *Topics in Cognitive Science*, n/a(n/a), 2021.

[62] Sarah A. Wu, Rose E. Wang, James A. Evans, Joshua B. Tenenbaum, David C. Parkes, 和 Max Kleiman-Weiner. 厨师太多：通过逆向规划协调多智能体协作。《认知科学专题》，无卷号(无期号)，2021年。

[63] Zhiyong Wu, Chengcheng Han, Zichen Ding, Zhenmin Weng, Zhoumianze Liu, Shunyu Yao, Tao Yu, and Lingpeng Kong. OS-copilot: Towards generalist computer agents with self-improvement. arXiv preprint arXiv:2402.07456, 2024.

[63] 吴志勇, 韩成成, 丁子辰, 翁振民, 刘周勉泽, 姚顺宇, 余涛, 和孔令鹏。OS-copilot: 迈向具备自我提升能力的通用计算机代理。arXiv预印本 arXiv:2402.07456, 2024年。

[64] Peter R. Wurman, Samuel Barrett, Kenta Kawamoto, James MacGlashan, Kaushik Subramanian, Thomas J. Walsh, Roberto Capobianco, Alisa Devlic, Franziska Eckert, Florian Fuchs, et al. Outracing champion Gran Turismo drivers with deep reinforcement learning. *Nature*, 602(7896):223-228, 2022.

[64] Peter R. Wurman, Samuel Barrett, Kenta Kawamoto, James MacGlashan, Kaushik Subramanian, Thomas J. Walsh, Roberto Capobianco, Alisa Devlic, Franziska Eckert, Florian Fuchs 等。用深度强化学习超越冠军《Gran Turismo》赛车手。《自然》, 602(7896):223-228, 2022年。

- [65] Yuchen Xiao, Weihao Tan, and Christopher Amato. Asynchronous actor-critic for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 35:4385-4400, 2022.
- [65] 肖宇晨, 谭伟豪, 和 Christopher Amato。多智能体强化学习中的异步演员-评论家方法。《神经信息处理系统进展》, 35:4385-4400, 2022年。
- [66] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. arXiv preprint arXiv:2404.07972, 2024.
- [66] 谢天宝, 张丹阳, 陈继轩, 李晓川, 赵思恒, 曹瑞升, 卓靖华, 程周俊, 申东灿, 雷方宇 等。Osworld: 在真实计算机环境中对多模态代理执行开放式任务的基准测试。arXiv预印本 arXiv:2404.07972, 2024年。
- [67] An Yan, Zhengyuan Yang, Wanrong Zhu, Kevin Lin, Linjie Li, Jianfeng Wang, Jianwei Yang, Yiwu Zhong, Julian McAuley, Jianfeng Gao, Zicheng Liu, and Lijuan Wang. GPT-4V in wonderland: Large multimodal models for zero-shot smartphone GUI navigation. arXiv preprint arXiv:2311.07562, 2023.
- [67] 安安, 杨正远, 朱万荣, 林凯文, 李林杰, 王建峰, 杨建伟, 钟义武, Julian McAuley, 高建峰, 刘子成, 和王丽娟。GPT-4V奇境: 用于零样本智能手机GUI导航的大型多模态模型。arXiv预印本 arXiv:2311.07562, 2023年。
- [68] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v, 2023.
- [68] 杨建伟, 张昊, 李峰, 邹雪艳, 李春元, 和高建峰。集合标记提示释放GPT-4V中卓越的视觉定位能力, 2023年。
- [69] Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. AppAgent: Multimodal agents as smartphone users. arXiv preprint arXiv:2312.13771, 2023.
- [69] 赵阳, 刘嘉轩, 韩宇成, 陈昕, 黄泽彪, 傅斌, 和余刚。AppAgent: 作为智能手机用户的多模态代理。arXiv预印本 arXiv:2312.13771, 2023年。
- [70] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744-20757, 2022.
- [70] 姚顺宇, 陈豪, 杨健, 和 Karthik Narasimhan。Webshop: 迈向具备落地语言代理的可扩展真实网络交互。《神经信息处理系统进展》, 35:20744-20757, 2022年。
- [71] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36, 2024.
- [71] 姚顺宇, 余点, 赵杰弗里, Izhak Shafran, Tom Griffiths, 曹元, 和 Karthik Narasimhan。思维树: 利用大型语言模型进行深思熟虑的问题解决。《神经信息处理系统进展》, 36, 2024年。
- [72] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.
- [72] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: 在语言模型中协同推理与行动。发表于国际学习表征会议 (ICLR), 2023年。
- [73] Chaoyun Zhang, Liqun Li, Shilin He, Xu Zhang, Bo Qiao, Si Qin, Minghua Ma, Yu Kang, Qingwei Lin, Saravan Rajmohan, et al. UFO: A UI-focused agent for Windows OS interaction. arXiv preprint arXiv:2402.07939, 2024.
- [73] Chaoyun Zhang, Liqun Li, Shilin He, Xu Zhang, Bo Qiao, Si Qin, Minghua Ma, Yu Kang, Qingwei Lin, Saravan Rajmohan, 等。UFO: 面向Windows操作系统交互的用户界面代理。arXiv预印本 arXiv:2402.07939, 2024年。

- [74] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. GPT-4V(ision) is a generalist web agent, if grounded. arXiv preprint arXiv:2401.01614, 2024.
- [74] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. GPT-4V(ision) 是一个通用的网络代理，前提是基础支持。arXiv预印本 arXiv:2401.01614, 2024年。
- [75] Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In ICLR, 2024.
- [75] Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: 基于轨迹示例的提示与记忆用于计算机控制。发表于ICLR, 2024年。
- [76] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, et al. WebArena: A realistic web environment for building autonomous agents. arXiv preprint arXiv:2307.13854, 2023.
- [76] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, 等。WebArena: 用于构建自主代理的真实网页环境。arXiv预印本 arXiv:2307.13854, 2023年。

## 21 Appendix

## 22 附录

Table of Contents

目录

A General Implementation 21

A 通用实现 21

B Red Dead Redemption II 23

B 荒野大镖客2 23

B.1 Introduction to RDR2 23

B.1 荒野大镖客2简介 23

B. 2 Objectives 23

B.2 目标 23

B.3 Implementation Details 25

B.3 实现细节 25

B.4 Case Studies 29

B.4 案例研究 29

B.5 Limitations of GPT-4o and GPT-4V 32

B.5 GPT-4o和GPT-4V的局限性 32

C Stardew Valley 35

C 星露谷物语 35

C. 1 Introduction to Stardew Valley 35

C.1 星露谷物语简介 35

C. 2 Objectives 35

C.2 目标 35

C. 3 Implementation Details 36

C. 3 实施细节 36

C. 4 Case Studies 38

C. 4 案例研究 38

C. 5 Limitations of GPT-4o 39

C. 5 GPT-4o的局限性 39

D Dealer's Life 2 39

D Dealer's Life 2 39

D. 1 Introduction to Dealer's Life 2 39

D. 1 Dealer's Life 2 简介 39

D. 2 Objectives 40

D. 2 目标 40

D. 3 Implementation Details 40

D. 3 实施细节 40

D. 4 Case Studies 41

D. 4 案例研究 41

D.5 Quantitative Evaluation 43

D.5 定量评估 43

D. 6 Evaluation Metrics 43

D. 6 评估指标 43

E Cities: Skylines 45

E Cities: Skylines 45

E.1 Introduction to Cities: Skylines 45

E.1 Cities: Skylines 简介 45

E.2 Objectives 46

E.2 目标 46

E.3 Evaluation Metric 46

E.3 评估指标 46

E. 4 Implementation Details 47

E. 4 实施细节 47

E.5 Case Studies 50

E.5 案例研究 50

F Software Applications 52

F 软件应用 52

F.1 Selected Software Applications 52

F.1 选定的软件应用 52

F. 2 Software Tasks 52

F.2 软件任务 52

F.3 Quantitative Evaluation 53

F.3 定量评估 53

F.4 Implementation Details 56

F.4 实现细节 56

F.5 Case Studies 60

F.5 案例研究 60

F. 6 Limitations of GPT-40 62

F.6 GPT-40的局限性 62

G OSWorld 63

G OSWorld 63

G.1 Introduction to OSWorld 63

G.1 OSWorld简介 63

G.2 OSWorld Tasks 63

G.2 OSWorld任务 63

G.3 Implementation Details 64

G.3 实现细节 64

G.4 Application Target and Setting Challenges 65

G.4 应用目标与设置挑战 65

G.5 Case Studies 66

G.5 案例研究 66

G.6 Quantitative Evaluation 66

G.6 定量评估 66

H Cradle Prompts 68

H Cradle提示 68

H. 1 Prompts for RDR2 68

H. 1 RDR2 的提示 68

H.2 Prompts for Cities: Skylines 78

H.2 Cities: Skylines 的提示 78

H. 3 Prompts for Stardew Valley 85

H. 3 Stardew Valley 的提示 85

H. 4 Prompts for Dealer's Life 2 104

H. 4 Dealer's Life 2 的提示 104

H. 5 Prompts for Software Applications 109

H. 5 软件应用的提示 109

## 23 A General Implementation

### 24 一般实现

Here we introduce the general implementation details of CRADLE. For specialized implementations addressing issues unique to their own environment, please refer to the corresponding section.

这里介绍 CRADLE 的一般实现细节。针对各自环境特有问题的专门实现，请参阅相应章节。

Backbone Model. We employ GPT-40 [43], currently one of the most capable LMM models, as the framework's backbone model. If not mentioned explicitly, all the experiments are done with gpt-40-2024-05-13. Temperature is set to 0 to lower the variance of the text generation.

骨干模型。我们采用 GPT-40 [43]，目前最强大的大型多模态模型（LMM）之一，作为框架的骨干模型。除非特别说明，所有实验均使用 gpt-40-2024-05-13。温度参数设为0，以降低文本生成的方差。

Observation Space. CRADLE only takes a video clip, which records the progress of execution of the last action, as input. To lower the frequency of interaction with backbone models and reduce the strain on the computer, video is recorded at 2 fps (a screenshot every 0.5 seconds), which proves to be sufficient in most cases for information gathering without missing any important information.

观测空间。CRADLE 仅以视频片段作为输入，该视频记录了上一次动作执行的进展。为降低与骨干模型的交互频率及减轻计算机负担，视频以2帧每秒录制（每0.5秒截屏一次），在大多数情况下足以收集信息且不遗漏重要内容。

Action Space. For the action space, it includes all possible keyboard and mouse operations, including key\_press, key\_hold, key\_release, mouse\_move, mouse\_click, mouse\_hold, mouse\_release, and wheel\_scroll, which can be combined in different ways to form combos and shortcuts, use keys in fast sequence, or coordinate timings. We choose to use Python code to simulate these operations and encapsulate them into an io\_env class. Skill code needs to be generated by the agent in order to utilize such functions and affordances so executed actions take effect. Table 7 illustrates CRADLE's action space.

动作空间。动作空间包括所有可能的键盘和鼠标操作，如按键按下 (key\_press)、按键保持 (key\_hold)、按键释放 (key\_release)、鼠标移动 (mouse\_move)、鼠标点击 (mouse\_click)、鼠标按住 (mouse\_hold)、鼠标释放 (mouse\_release) 和滚轮滚动 (wheel\_scroll)，这些操作可组合形成连招和快捷键，快速连续使用按键，或协调时序。我们选择用 Python 代码模拟这些操作，并封装成 io\_env 类。代理需生成技能代码以利用这些功能和操作，从而使执行的动作生效。表7展示了 CRADLE 的动作空间。

Table 7: Action space in the CRADLE framework, including action attributes. Coordinate system is either absolute or relative. Actions with durations can be either synchronous or asynchronous

表7：CRADLE 框架中的动作空间及动作属性。坐标系可为绝对或相对。带持续时间的动作可为同步或异步

Type	Action	Attributes
	Key Press	Key name (string), Key press duration (seconds:float)
	Key Hold	Key name (string)
	Key Release	Key name (string)
Keyboard	Key Combo	Key names (strings), Key combo duration (seconds:float), Wait behaviour (sync/async)
	Hotkey	Key names (strings), Hotkey sequence duration (seconds:float), Wait behaviour (sync/async)
	Text Type	String to type (string), Typing duration (seconds:float)
	Button Click	Mouse button (left/middle/right), Button click duration (seconds:float)
	Button Hold	Mouse button (left/middle/right)
	Button Release	Mouse button (left/middle/right)
Mouse	Move	Mouse position (width:int, height:int), Mouse speed (seconds:float), Coordinate system (relative/absolute), Tween mode (enum) \${}^{\{3\}}\$
	Scroll	Orientation (vertical), Distance (pixels:int), Duration (seconds:float)
Wait	Noop	-
类型	动作	属性
	按键按下	按键名称（字符串）， 按键按下持续时间（秒：浮点数）
	按键保持	按键名称（字符串）
	按键释放	按键名称（字符串）
键盘	按键组合	按键名称（字符串）， 按键组合持续时间（秒：浮点数）， 等待行为（同步/异步）
	快捷键	按键名称（字符串）， 快捷键序列持续时间（秒：浮点数）， 等待行为（同步/异步）
	文本输入	输入字符串（字符串）， 输入持续时间（秒：浮点数）
	按钮点击	鼠标按钮（左/中/右）， 按钮点击持续时间（秒：浮点数）
	按钮保持	鼠标按钮（左/中/右）
	按钮释放	鼠标按钮（左/中/右）
鼠标	移动	鼠标位置（宽度：整数，高度：整数）， 鼠标速度（秒：浮点数）， 坐标系（相对/绝对）， 缓动模式（枚举）\${}^{\{3\}}\$
	滚动	方向（垂直）， 距离（像素：整数）， 持续时间（秒：浮点数）
等待	无操作	-

It is important to note that, while some works (e.g., AssistantGUI [18], OmniACT [27] and OS-World [66]) use PyAutoGUI<sup>4</sup> for keyboard and mouse control, this approach does not work in all applications, particularly in modern video games using DirectX<sup>5</sup>. Moreover, such work chooses to

需要注意的是，虽然一些工作（例如 AssistantGUI [18]、OmniACT [27] 和 OS-World [66]）使用 PyAutoGUI<sup>4</sup> 进行键盘和鼠标控制，但这种方法并不适用于所有应用，尤其是使用 DirectX<sup>5</sup> 的现代视频游戏。此外，这些工作选择

<sup>4</sup> Python library that provides a cross-platform GUI automation module - <https://github.com/asweigart/pyautogui>

<sup>4</sup> 提供跨平台 GUI 自动化模块的 Python 库 - <https://github.com/asweigart/pyautogui>

<sup>5</sup> Microsoft DirectX graphics provides a set of APIs for high-performance multimedia apps - <https://learn.microsoft.com/en-us/windows/win32/directx>

<sup>5</sup> 微软 DirectX 图形提供了一套用于高性能多媒体应用的 API - <https://learn.microsoft.com/en-us/windows/win32/directx>

---

expose a subset of the library functionality in its action space, ignoring dimensions like press duration and movement speed, which are critical in many scenarios (e.g., RDR2, for opening the weapon wheel and changing view).

在其动作空间中仅暴露库功能的子集，忽略了按键持续时间和移动速度等维度，而这些在许多场景中（例如 RDR2 中打开武器轮盘和切换视角）至关重要。

To ensure wide game and software compatibility and accommodate different operating systems, in our current implementation we use the similar PyDirectInput library <sup>6</sup> and PyAutoGUI for keyboard control, utilize *AHK*<sup>7</sup> and write our own abstraction (using the ctypes library <sup>8</sup>) to send low-level mouse commands to the operating system for mouse control. For increased portability and ease of maintenance, all keyboard and mouse control is encapsulated in a class, called IO\_env.

为了确保广泛的游戏和软件兼容性并适应不同操作系统，在我们当前的实现中，我们使用类似的 PyDirectInput 库 <sup>6</sup> 和 PyAutoGUI 进行键盘控制，利用 *AHK*<sup>7</sup> 并编写自己的抽象层（使用 ctypes 库 <sup>8</sup>）向操作系统发送低级鼠标命令以实现鼠标控制。为了提高可移植性和维护便利性，所有键盘和鼠标控制都封装在一个名为 IO\_env 的类中。

Notably, our low-level control wrapper is adapted for both MacOS and Windows systems, making the OS transparent to us. At the software window level, we implemented automatic switching between the target software window and the window running the agent (using Python ctypes for Windows and AppleScript for MacOS <sup>9</sup>).  
值得注意的是，我们的低级控制封装适配了 MacOS 和 Windows 系统，使操作系统对我们透明。在软件窗口层面，我们实现了目标软件窗口与运行代理窗口之间的自动切换（Windows 使用 Python ctypes，MacOS 使用 AppleScript <sup>9</sup>）。

Procedure Memory. This memory stores pre-defined basic skills and the generated skills captured from the Skill Curation. However, as we continuously obtain new skills during game playing, the number of skills in procedural memory keeps increasing, and it is hard for GPT-40 to precisely select the most suitable skill from the large memory. Thus, similar to Voyager [57], we use OpenAI's text-embedding-ada-002 model [42] to generate embeddings for each skill and store pre-defined basic skills and any generated skills captured from Skill Curation, along with their embeddings in a procedural memory. We retrieve a subset of skills, that are relevant to the given task, and then let GPT-40 select the most suitable one from the subset. In the skill retrieval, we pre-compute the embeddings of the documentations (code, comments and descriptions) of skill functions, which describe the skill functionality, and compute the embedding of the given task. Then we compute the cosine similarities between the skill documentation embeddings and the task embedding. The higher similarity means that the skill's functionality is more relevant to the given task. We select the top K skills with the highest similarities as the subset. Using similarity matching to select a small candidate set simplifies the process of choosing skills.

过程记忆。该记忆存储预定义的基础技能和从技能策划中捕获的生成技能。然而，随着游戏进行中不断获得新技能，过程记忆中的技能数量持续增加，GPT-40 很难从庞大的记忆中精确选择最合适的技能。因此，类似于 Voyager [57]，我们使用 OpenAI 的 text-embedding-ada-002 模型 [42] 为每个技能生成嵌入，并将预定义基础技能及从技能策划捕获的生成技能连同其嵌入存储在过程记忆中。我们检索与给定任务相关的技能子集，然后让 GPT-40 从该子

集中选择最合适的技能。在技能检索中，我们预先计算技能函数文档（代码、注释和描述）的嵌入，这些文档描述了技能功能，并计算给定任务的嵌入。随后计算技能文档嵌入与任务嵌入之间的余弦相似度。相似度越高，说明技能功能与给定任务越相关。我们选择相似度最高的前 K 个技能作为子集。使用相似度匹配选择小的候选集简化了技能选择过程。

Episodic Memory. This memory stores all the useful information provided by the environment and LMM, which consists of short-term memory and long-term summary.

情景记忆。该记忆存储环境和大规模语言模型（LMM）提供的所有有用信息，包含短期记忆和长期摘要。

The short-term memory stores the screenshots within the recent k interactions in game playing and the corresponding information from other modules, e.g., screenshot descriptions, task guidance, actions, and reasoning. We set k to five, and it can be regarded as the memory length. Information stored over k interactions ago will be forgotten from direct short-term memory. Empirically, we found that recent information is crucial for decision-making, while a too-long memory length would cause hallucinations. In addition, other modules continuously retrieve recent information from short-term memory and update the short-term memory by storing the newest information.

短期记忆存储最近 k 次游戏交互中的截图及其他模块对应的信息，例如截图描述、任务指导、动作和推理。我们设置 k 为五，可视为记忆长度。超过 k 次交互之前的信息将从直接短期记忆中遗忘。经验表明，近期信息对决策至关重要，而过长的记忆长度会导致幻觉。此外，其他模块持续从短期记忆中检索近期信息，并通过存储最新信息更新短期记忆。

For some long-horizon tasks, short-term memory is not enough. This is because the completion of a long-horizon task might require historical information from a long time ago. For example, the agent might do a series of short-horizon tasks during a long-horizon task, which makes the original long-horizon task forgotten in short-term memory. To maintain the long-term valuable information while avoiding the long-token burden of GPT-4o, we propose a recurrent information summary as long-term memory, which is the text summarization of experiences in game playing, including the ongoing task, the past entities that the player met, and the past behaviors of the player and NPCs.

对于一些长时间跨度的任务，短期记忆不足以应对。这是因为完成长时间跨度任务可能需要很久之前的历史信息。例如，代理可能在执行长时间跨度任务期间完成一系列短时间跨度任务，这会导致原始长时间跨度任务在短期记忆中被遗忘。为保持长期有价值的信息，同时避免 GPT-40 的长令牌负担，我们提出了递归信息摘要作为长期记忆，即对游戏体验的文本总结，包括当前任务、玩家遇到的过去实体以及玩家和 NPC 的过去行为。

In more detail, we provide GPT-4o with the summarization before the current screenshot and the recent screenshots with corresponding descriptions, and GPT-4o will make a new summarization by organizing the tasks, entities, and behaviors in the time order with sentence number restriction. Then we update the summarization to be the newly generated one, which includes the information in

更具体地，我们在当前截图之前向 GPT-40 提供摘要和最近截图及其对应描述，GPT-40 将通过按时间顺序组织任务、实体和行为并限制句子数来生成新的摘要。然后我们将摘要更新为新生成的版本，其中包含

---

<sup>6</sup> Python library encapsulating Microsoft's DirectInput calls for convenience manipulating keyboard keys - <https://github.com/learncodebygaming/pydirectinput>

<sup>6</sup> 封装微软DirectInput调用的Python库，方便操作键盘按键 - <https://github.com/learncodebygaming/pydirectinput>

<sup>7</sup> A fully typed Python wrapper around AutoHotkey to keyboard and mouse control - <https://github.com/spyoungtech/ahk>

<sup>7</sup> 基于AutoHotkey的全类型Python封装，用于键盘和鼠标控制 - <https://github.com/spyoungtech/ahk>

<sup>8</sup> Python library that provides C compatible data types, and allows calling functions in DLL/.so binaries - <https://docs.python.org/3/library/ctypes.html>

<sup>8</sup> 提供C兼容数据类型并允许调用DLL/.so二进制函数的Python库 - <https://docs.python.org/3/library/ctypes.html>

<sup>9</sup> AppleScript is a scripting language created by Apple, which allows users to directly control scriptable applications, as well as parts of MacOS - [https://developer.apple.com/library/archive/documentation/AppleScript/Conceptual/AppleScriptLangGuide/introduction/ASLR\\_intro.html](https://developer.apple.com/library/archive/documentation/AppleScript/Conceptual/AppleScriptLangGuide/introduction/ASLR_intro.html)

<sup>9</sup> AppleScript是苹果公司开发的一种脚本语言，允许用户直接控制可脚本化应用程序及MacOS部分功能 - [https://developer.apple.com/library/archive/documentation/AppleScript/Conceptual/AppleScriptLangGuide/introduction/ASLR\\_intro.html](https://developer.apple.com/library/archive/documentation/AppleScript/Conceptual/AppleScriptLangGuide/introduction/ASLR_intro.html)

---

the current screenshot. The recurrent summarization update, inspired by RNN, achieves linear-time inference by preserving a hidden state that encapsulates historical input. This method ensures the compactness of summarization token lengths and recent input data. Furthermore, the incorporation of long-term memory enables the agent to effectively retain crucial information over extended periods, thereby enhancing decision-making capabilities.

当前截图。受循环神经网络（RNN）启发的递归摘要更新，通过保留封装历史输入的隐藏状态，实现线性时间推理。该方法确保摘要标记长度和近期输入数据的紧凑性。此外，长时记忆的引入使代理能够有效保留关键长时间信息，从而增强决策能力。

Information Gathering. Given the video clip as input, we mainly depend on GPT-4o's OCR capabilities to extract textual information in the keyframes, which usually contain critical guidance and notifications for the current situation. We also rely on GPT-4o's visual understanding to analyze the visual information in the frames. Besides, we augment LMMs' visual understanding via some tools, like template matching [8], Grounding DINO [35], and SAM [29], to provide additional grounding for object detection and segmentation. Some visual prompting tricks, like drawing axes and colorful directional bands, are also applied to enhance the GPT-4o's visual ability.

信息收集。以视频片段为输入，我们主要依赖GPT-4o的OCR能力提取关键帧中的文本信息，这些信息通常包含当前情境的重要指引和通知。我们还依赖GPT-4o的视觉理解分析帧中的视觉信息。此外，我们通过一些工具如模板匹配[8]、Grounding DINO[35]和SAM[29]增强大模型（LMMs）的视觉理解，为目标检测和分割提供额外的定位支持。还应用了一些视觉提示技巧，如绘制坐标轴和彩色方向带，以提升GPT-4o的视觉能力。

Task Inference. After reflecting on the outcome of the last executed action, We let GPT-4o analyze the current situation to infer the most suitable task for the current moment and estimate the highest priority task to perform and when to stop an ongoing task and start a new one.

任务推断。在反思上一次执行动作的结果后，我们让GPT-4o分析当前情境，推断当前最合适的任务，评估执行优先级最高的任务，以及何时停止正在进行的任务并开始新任务。

Skill Curation. GPT-40 is required to strictly follow the provided interfaces and examples to generate the corresponding code for new skills. Moreover, GPT-40 is required to include documentation/comments within the generated code, delineating the functionality of each skill. Procedural Memory where skills are stored will then check whether the code is valid, whether the format of documentation is right, and whether any skill with the same name already exists. If all conditions are passed, the newly generated skill is persisted for future utilization.

技能策划。要求GPT-40严格遵循提供的接口和示例生成新技能的对应代码。此外，GPT-40需在生成代码中包含文档/注释，说明每个技能的功能。存储技能的过程将检查代码有效性、文档格式是否正确及是否存在同名技能。若所有条件满足，则新生成的技能将被保存以供后续使用。

Action Planning. GPT-40 needs to select the appropriate skills from the curated skill set and instantiate these skills into a sequence of executable actions by specifying any necessary parametric aspects (e.g., duration, position, and target) according to the current task and history information. The generated action is then fed to the Executor for interaction with the environment.

动作规划。GPT-40需从策划的技能集中选择合适技能，并根据当前任务和历史信息，通过指定必要的参数（如持续时间、位置和目标）将这些技能实例化为可执行动作序列。生成的动作随后交由执行器与环境交互。

## 25 B Red Dead Redemption II

### 26 B 荒野大镖客救赎2

#### 26.1 B.1 Introduction to RDR2

#### 26.2 B.1 RDR2简介

Red Dead Redemption II (RDR2) is an epic AAA Western-themed action-adventure game by Rockstar Games. As one of the most famous and highest-selling games in the world, it is widely acknowledged for its movie-like realistic scenes, rich storylines, and immersive open-ended world. The game applies a typical role-playing game (RPG) control system, played from a first- or third-person perspective, which uses WASD for movement, mouse control for view changing, first- or third-person shooting for combat, and inventory and manipulation.

《荒野大镖客救赎2》(Red Dead Redemption II, 简称RDR2)是Rockstar Games出品的一款史诗级AAA西部题材动作冒险游戏。作为全球最著名且销量最高的游戏之一，它以电影般逼真的场景、丰富的剧情和沉浸式开放世界著称。游戏采用典型的角色扮演游戏(RPG)控制系统，支持第一人称或第三人称视角，使用WASD键移动，鼠标控制视角切换，第一或第三人称射击进行战斗，以及物品栏和操作管理。

For most of the game, players need to control the main character, Arthur Morgan, upon choosing to complete mission scenarios following the main storyline. Otherwise, they can freely explore the interactive world, such as going hunting, fishing, chatting with non-player characters (NPCs), training horses, witnessing or partaking in random events, and participating in side quests. As the main storyline progresses, different skills are gradually unlocked. As a close-source commercial game, no APIs are available for obtaining additional game-internal information nor pre-defined automation actions. Following its characteristics, this game serves as a fitting and challenging environment for the GCC setting and a comprehensive benchmark for embodiment.

游戏大部分时间，玩家需控制主角亚瑟·摩根(Arthur Morgan)，选择完成主线剧情任务。否则，玩家可自由探索互动世界，如狩猎、钓鱼、与非玩家角色(NPC)交谈、训练马匹、目睹或参与随机事件及支线任务。随着主线推进，不同技能逐步解锁。作为闭源商业游戏，无法通过API获取额外游戏内部信息或预定义自动化操作。基于其特性，该游戏为GCC环境提供了合适且具挑战性的测试场景，也是体现能力的综合基准。

#### 26.3 B.2 Objectives

#### 26.4 B.2 目标

In Chapter 1 of RDR2, the first two missions of the main storyline are Outlaws from the West and Enter, Pursued by a Memory. These missions serve as the tutorial content for RDR2, guiding players step-by-step into the role of Arthur. They immerse the player in the story's development while teaching the game's controls and mechanics.

在RDR2第一章中，主线的前两个任务分别是《西部亡命徒》和《进入，被记忆追逐》。这些任务作为RDR2的教程内容，逐步引导玩家进入亚瑟的角色，沉浸于故事发展，同时教授游戏的操作和机制。

We divided Mission 1 and Mission 2 into 8 and 5 tasks respectively based on the checkpoints within each mission. Each checkpoint may present failure scenarios. For example, in Mission 1, there are six failure scenarios: i) Assaults, kills, or abandons Dutch or Micah; ii) Allows Dutch or Micah to be killed; iii) Abandons the homestead; iv) Assaults, kills, or abandons their horse; v) Assaults, kills, or

我们根据每个任务中的检查点，将任务1和任务2分别划分为8个和5个子任务。每个检查点可能会出现失败场景。例

如，在任务1中，有六种失败场景：i) 攻击、杀害或遗弃Dutch或Micah；ii) 允许Dutch或Micah被杀；iii) 遗弃家园；iv) 攻击、杀害或遗弃他们的马；v) 攻击、杀害或

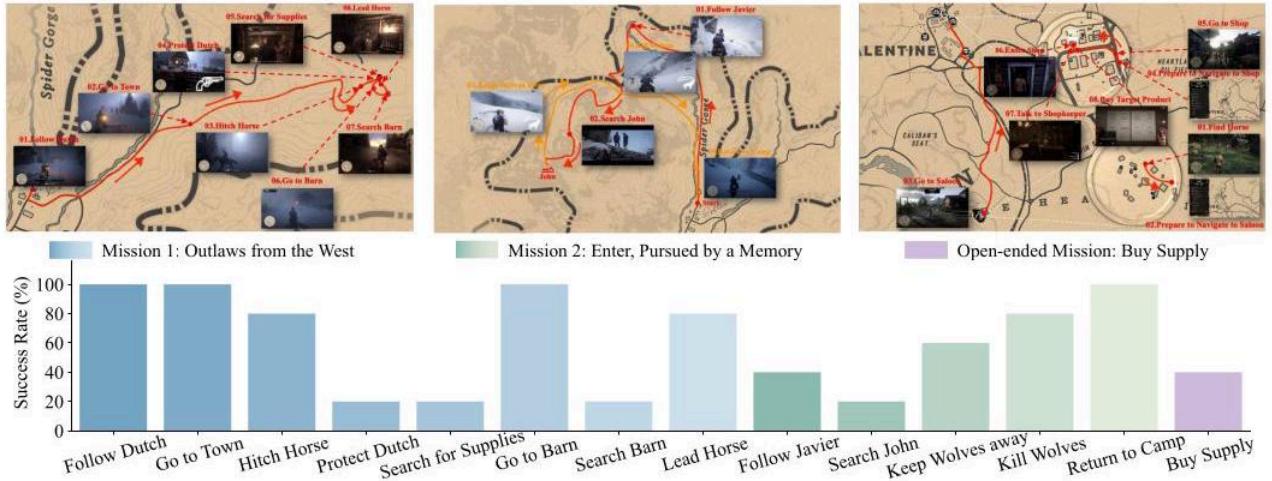


Figure 12: Trajectory and success rates of 13 main storyline tasks and 1 open-world task in RDR2.

图12: RDR2中13个主线任务和1个开放世界任务的轨迹及成功率。

abandons the horse in the barn; vi) Dies. We categorized each sub-task as either "Easy" or "Hard" based on the likelihood of failure at each checkpoint and the need to retry the checkpoint.

遗弃马厩中的马；vi) 死亡。我们根据每个检查点失败的可能性及是否需要重试，将每个子任务分类为“简单”或“困难”。

To evaluate CRADLE's capabilities in an open-world environment, Mission 3 is designed as a hard open-ended task. Unlike the first two tutorial missions, it does not include any checkpoints. Consequently, the entire Mission 3 is treated as a single, comprehensive task. Although we do not subdivide Mission 3 into finer tasks, we aim to identify key points to facilitate a clearer understanding of Mission 3 for the reader.

为了评估CRADLE在开放世界环境中的能力，任务3被设计为一个困难的开放式任务。与前两个教程任务不同，它不包含任何检查点。因此，整个任务3被视为一个完整的综合任务。虽然我们没有将任务3细分为更小的任务，但我们旨在识别关键点，以便读者更清晰地理解任务3。

Tables 8 and 9 provide a brief introduction of each task in the first two missions of the main storyline and an open-ended mission, along with approximate estimates of their difficulty. Due to GPT-40's poor performance in spatial understanding and fine-manipulation skills, it can be challenging for our agent to perform certain actions, like entering or leaving a building, or going to precise indoor locations to retrieve specific items. Additionally, the high latency of GPT-40's responses also makes it harder for an agent to deal with time-sensitive events, e.g., during combat.

表8和表9简要介绍了主线前两个任务和一个开放式任务中的各个子任务，并给出了难度的近似估计。由于GPT-40在空间理解和精细操作技能方面表现较差，我们的代理在执行某些动作时可能会遇到困难，比如进入或离开建筑物，或前往精确的室内位置取回特定物品。此外，GPT-40响应的高延迟也使代理更难应对时间敏感事件，例如战斗期间。

Table 8: Tasks in the first two missions of RDR2. In the tutorial guide, the prompt text Start Dialogue signifies the end of the previous checkpoint and the beginning of the current checkpoint. Difficulty refers to how hard to accomplish the corresponding tasks. Figures 13 and 14 showcase snapshots of each task (specific sub-figures marked in parenthesis in the table). The maximal number of steps (agent takes one action) for each task is 500.

表8: RDR2前两个任务中的子任务。在教程指南中，提示文本“开始对话”表示前一个检查点结束和当前检查点开始。难度指完成相应任务的难易程度。图13和图14展示了每个任务的快照（表中括号内标注具体子图）。每个任务的最大步骤数（代理执行一次动作）为500步。

Mission 1: Outlaws from the West	Description	Start Dialogue	Difficulty
Follow Dutch (Fig. 13a)	Arthur follows Dutch on horseback into the snow to find their scouting gang members.	Use [W] to Follow Dutch	Easy
Go to Town (Fig. 13b)	Arthur rides his horse, following Micah to the vicinity of a little homestead Micah discovered.	Hold [W] to match speed with Dutch and Micah	Easy
Hitch Horse (Fig. 13c)	Arthur hitches the horse to the hitching post, then goes to the old shed and takes cover.	Hold [E] to hitch your horse	Easy
Protect Dutch (Fig. 13d)	Arthur uses his gun to shoot all of the O'Driscolls inhabiting the house and protect Dutch.	Use [W] to peak out of cover	Hard
Search for Supplies (Fig. 13e)	Arthur follows Dutch to the house to search for supplies.	Hold [R] near items to pick them up while searching house.	Hard
Go to Barn (Fig. 13f)	Arthur follows Dutch's directions and goes to the barn to see if there's anything inside.	Dutch: Micah, Arthur, keep looking for stuff	Easy
Search Barn (Fig. 13g)	Arthur searches the barn and defeats the O'Driscoll hiding inside.	[F] Attack the O'Driscoll	Hard
Lead Horse (Fig. 13h)	Arthur calms the horse and takes it out of the barn.	Hold [Right Mouse Button] to focus on the horse	Easy
Mission 2: Enter, Pursued by a Memory	Description	Start Dialogue	Difficulty
Follow Javier (Fig. 14a)	Arthur rides his horse following Javier up the mountain through the blizzard searching for John's trail.	Follow Javier	Hard
Search John (Fig. 14b)	After dismounting, Arthur followed Javier over slopes and ledges to find John and carry him away.	Javier: Down this way	Hard
Keep Wolves away (Fig. 14c)	Arthur manages to shoot all of the wolves before they can attack Javier and John.	Keep the wolves away from Javier and John	Hard
Kill Wolves (Fig. 14d)	Three people ride horses down the mountain. Arthur eliminate the wolves, protecting Javier and John ahead.	Javier: Come on, let's get back to the others	Hard
Return to Camp (Fig. 14e)	Arthur followed Javier on horseback back to camp.	Yea...c'mon. Let's push hard and get back	Easy

任务1: 来自西部的亡命之徒	描述	开始对话	难度
跟随Dutch (图13a)	Arthur骑马跟随Dutch进入雪地, 寻找他们的侦察队成员。	使用[W]跟随Dutch	简单
前往城镇 (图13b)	Arthur骑马跟随Micah来到Micah发现的小农舍附近。	按住[W]以与Dutch和Micah保持同速	简单
拴马 (图13c)	Arthur将马拴在拴马桩上, 然后走到旧棚子里躲藏。	按住[E]拴马	简单
保护Dutch (图13d)	Arthur用枪射击屋内所有O'Driscoll帮派成员, 保护Dutch。	使用[W]从掩体中探出头	困难
搜寻补给 (图13e)	Arthur跟随Dutch到房子里搜寻补给。	靠近物品时按住[R]拾取, 搜寻房屋时使用	困难
前往谷仓 (图13f)	Arthur按照Dutch的指示前往谷仓, 查看里面是否有物品。	Dutch: Micah, Arthur, 继续找东西	简单
搜查谷仓 (图13g)	Arthur搜查谷仓并击败藏在里面的O'Driscoll帮派成员。	[F] 攻击O'Driscoll	困难
牵马 (图13h)	Arthur安抚马匹, 将其牵出谷仓。	按住[鼠标右键]聚焦马匹	简单
任务2: 进入, 被记忆追逐	描述	开始对话	难度
跟随Javier (图14a)	Arthur骑马跟随Javier穿越暴风雪上山, 寻找John的踪迹。	跟随Javier	困难
寻找John (图14b)	下马后, Arthur跟随Javier穿过山坡和悬崖, 找到John并将他带走。	Javier: 这边走	困难
驱赶狼群 (图14c)	Arthur成功射杀所有狼群, 防止它们攻击Javier和John。	保护Javier和John, 驱赶狼群	困难
击杀狼群 (图14d)	三人骑马下山, Arthur消灭狼群, 保护前方的Javier和John。	Javier: 快点, 我们回去找其他人	困难
返回营地 (图14e)	Arthur骑马跟随Javier返回营地。	是的.....快点。我们加把劲回去	简单

Table 9: Key points in the open-ended mission, Buy Supply in RDR2. Figure 15 showcases snapshots of key points (specific sub-figures marked in parenthesis in the table).

表9: RDR2中开放式任务“购买补给”的关键点。图15展示了关键点的快照（表中括号内标注了具体子图）。

Mission 3: Buy Supply	Description
Find Horse (Fig. 15a)	Find and mount the horse in the camp.
Prepare to Navigate to Saloon (Fig. 15b)	Open map, find the saloon and create waypoint.
Go to Saloon (Fig. 15c)	Ride horse to the saloon.
Prepare to Navigate to Shop (Fig. 15d)	Open map, find the general store and create waypoint.
Go to Shop (Fig. 15e)	Ride horse to the shop.
Enter Shop (Fig. 15f)	Dismount the horse and enter the shop.
Talk to Shopkeeper(Fig. 15g)	Approach the shopkeeper and talk.
Buy Target Product (Fig. 15h)	Open the menu, find and buy the target product.

任务3: 购买物资	描述
找到马匹 (图15a)	在营地找到并骑上马。
准备导航到酒馆 (图15b)	打开地图, 找到酒馆并创建路径点。
前往酒馆 (图15c)	骑马前往酒馆。
准备导航到商店 (图15d)	打开地图, 找到杂货店并创建路径点。
前往商店 (图15e)	骑马前往商店。
进入商店 (图15f)	下马并进入商店。
与店主交谈 (图15g)	走近店主并交谈。
购买目标产品 (图15h)	打开菜单, 找到并购买目标产品。



Figure 13: Image examples of tasks in the first mission of Outlaws from the West. (The picture has been brightened for easier reading.)

图13：西部亡命徒第一任务中的图像示例。 (图片已调亮以便阅读。)

## 26.5 B.3 Implementation Details

### 26.6 B.3 实现细节

Our experiments are based on the latest version of RDR2, 'Build 1491.50'. As shown in Figure 16, strictly following the GCC setting, our agent takes the video of the screen as input and outputs keyboard and mouse operations to interact with the computer and the game. An observation thread is responsible for the collection of video frames from the screen and each video clip records the whole in-game process since executing the last action.

我们的实验基于最新版本的《荒野大镖客2》（RDR2），“Build 1491.50”。如图16所示，严格遵循GCC设置，我们的代理以屏幕视频作为输入，输出键盘和鼠标操作以与计算机和游戏交互。一个观察线程负责从屏幕采集视频帧，每个视频片段记录自上次执行动作以来的整个游戏过程。

Information Gathering. To extract keyframes from the video observation, we utilize the VideoSub-Finder tool<sup>10</sup>, a professional subtitle discovery and extraction tool. These keyframes usually contain rich meaningful textual information in the game, which are highly relevant to the completion of tasks and missions (such as character status, location, dialogues, in-game prompts and tips, etc.) We use GPT-4o to extract and categorize all the meaningful contexts in these keyframes and perform OCR, and call this processing "gathering text information". Then, to save interactions with GPT-4o, we only let GPT-4o provide a detailed description of the last frame of the video.

信息收集。为了从视频观察中提取关键帧，我们使用了VideoSub-Finder工具<sup>10</sup>，这是一款专业的字幕发现和提取工具。这些关键帧通常包含游戏中丰富且有意义的文本信息，与任务和使命的完成高度相关（如角色状态、位置、对话、游戏内提示等）。我们使用GPT-4o提取并分类这些关键帧中的所有有意义内容并执行OCR，将此过程称为“收集文本信息”。随后，为节省与GPT-4o的交互，我们仅让GPT-4o对视频的最后一帧提供详细描述。

While GPT-40 exhibits impressive visual understanding abilities across various CV tasks, we find that it struggles with spatial reasoning and recognizing some game-specific icons. To address these

尽管GPT-4o在多种计算机视觉任务中展现了出色的视觉理解能力，但我们发现它在空间推理和识别某些游戏特定图标方面存在困难。为了解决这些

<sup>10</sup> VideoSubFinder standalone tool - <https://sourceforge.net/projects/videosubfinder/>

<sup>10</sup> VideoSubFinder独立工具 - <https://sourceforge.net/projects/videosubfinder/>



Figure 14: Image examples of tasks in the second mission of Enter, Pursued by a Memory.

图14：进入“被记忆追逐”第二任务中的图像示例。

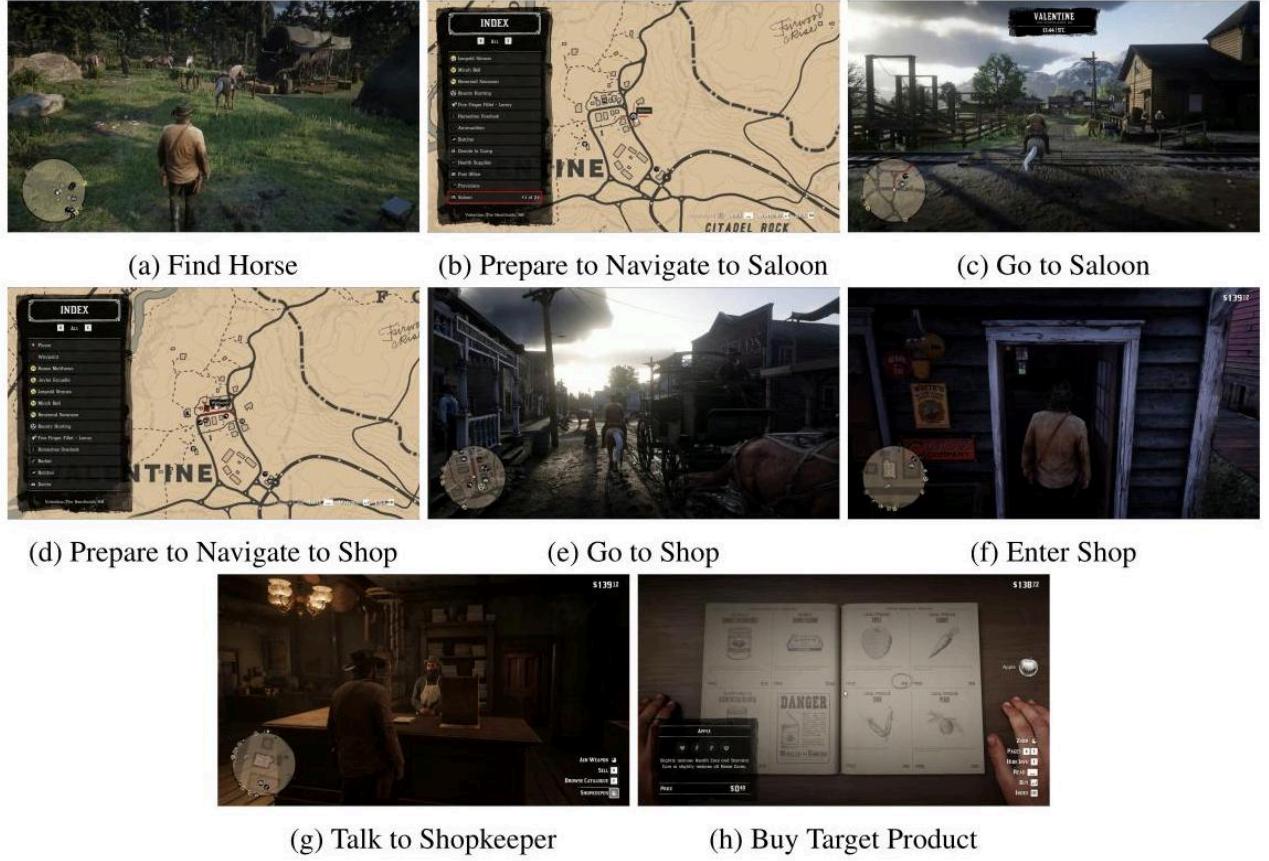


Figure 15: Image examples of key points in the open-ended task of Buy Supply.

图15：购买补给开放式任务关键点的图像示例。

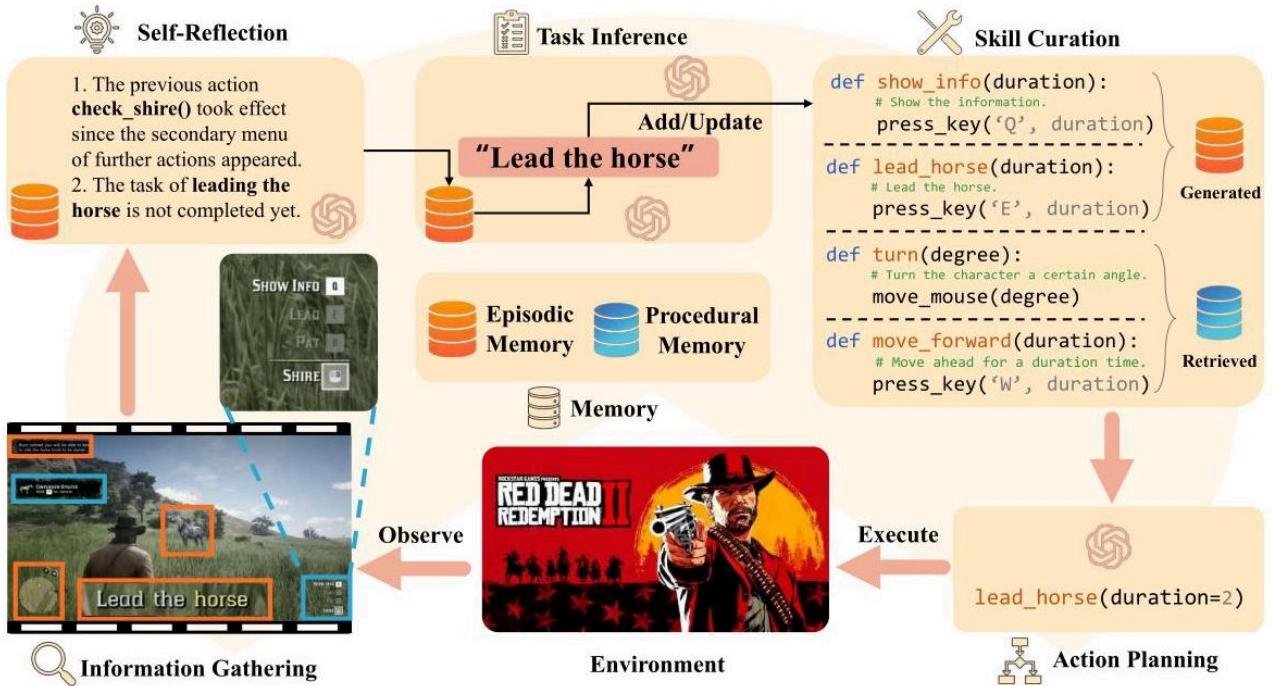


Figure 16: The detailed illustration of how CRADLE is instantiated as a game agent to play RDR2.

图16：CRADLE作为游戏代理玩《荒野大镖客2》的详细示意图。

limitations, we add a visual augmentation sub-module within our Information Gathering module. This augmentation step serves two main purposes: i) utilize Grounding DINO [35], an open-set object detector, to output precise bounding boxes of possible targets in an image and serve as spatial clues for GPT-4o; and ii) perform template matching [8] to provide icon recognition grounding truth for GPT-4o when interpreting instructions or menus shown on screen. As LMM capabilities mature, it should be possible to disable such augmentation.

限制，我们在信息收集模块中增加了视觉增强子模块。该增强步骤主要有两个目的：一是利用Grounding DINO [35]，一种开放集目标检测器，输出图像中可能目标的精确边界框，作为GPT-4o的空间线索；二是执行模板匹配 [8]，为GPT-4o在解释屏幕上显示的指令或菜单时提供图标识别的标注真值。随着大规模多模态模型（LMM）能力的成熟，未来应可关闭此类增强。

**Self-Reflection.** The reflection module mainly serves to evaluate whether the previously executed action was successfully carried out and whether the current executing task is finished. To achieve this, we uniformly sample at most 8 sequential frames from the video observation since the execution of the last action and use GPT-4o to estimate the success of its execution. Additionally, we expect GPT-4o can also provide analysis for any failure of the last action (e.g., the move-forward action failed and the cause could be the agent was blocked by an obstacle). With such valuable information as input for Action Planning, including the failure/success of the last action and the corresponding analysis, the agent is capable of attempting to remedy an inappropriate decision or action execution.

自我反思。反思模块主要用于评估先前执行的动作是否成功完成，以及当前执行的任务是否已完成。为此，我们从自上次动作执行以来的视频观察中均匀采样最多8帧连续图像，使用GPT-4o评估动作执行的成功与否。此外，我们期望GPT-4o还能对上次动作失败的原因进行分析（例如，前进动作失败，可能是代理被障碍物阻挡）。利用这些宝贵信息作为动作规划的输入，包括上次动作的成败及相应分析，代理能够尝试纠正不当的决策或动作执行。

Moreover, some actions require prolonged durations, such as holding down specific keys, which can coexist or interfere with other actions decided by subsequent decisions. Consequently, the reflection module must also decide whether an ongoing action should continue to be executed. Furthermore, self-reflection can be leveraged to dissect why the last action failed to bring the agent close to the target task completion, better understand the factors that led to the successful completion of the preceding task, and so on.

此外，某些动作需要较长时间持续执行，如按住特定按键，这些动作可能与后续决策决定的其他动作共存或产生干扰。因此，反思模块还必须决定是否应继续执行正在进行的动作。进一步地，自我反思可用于剖析为何上次动作未能使代理接近目标任务完成，更好地理解导致前一任务成功完成的因素等。

Besides, we observe that instead of providing GPT-4o with sequential high-resolution images for self-reflection, low-resolution images make it easier for GPT-4o to understand the relation among the sequential screenshots and capture dynamic changes, resulting in a significantly higher success rate of detecting whether the action is executed successfully and take any effect. We hypothesize that since a high-resolution image can cost as many as 2000 tokens, too many high-resolution images make GPT-4o fail to capture the overall changes across screenshots and be caught up in the local details.

此外，我们观察到，与向GPT-4o提供连续高分辨率图像进行自我反思相比，低分辨率图像更易于GPT-4o理解连续截图之间的关系并捕捉动态变化，从而显著提高检测动作是否成功执行及产生效果的成功率。我们推测，高分辨率图像可能消耗多达2000个token，过多高分辨率图像使GPT-4o难以捕捉截图间的整体变化，反而陷入局部细节。

**Task Inference.** During gameplay, we let GPT-4o propose the current task to perform whenever it believes it is time to start a new task. GPT-4o also outputs whether the task is a long- or short-horizon task when proposing a new task. Long-horizon tasks, such as traveling to a location, typically require multiple iterations, whereas short-horizon tasks, like picking up an item or conversing with someone, involve fewer iterations. The agent will follow the newly generated task for the next 3 interactions. After 3 interactions, the agent returns to the last long-horizon task in the stack. Deciding on a binary task horizon is much easier and more robust for GPT-4o, than re-planning at every iteration. Since

任务推断。在游戏过程中，每当GPT-4o认为是时候开始新任务时，会提出当前应执行的任务。GPT-4o还会在提出新任务时输出该任务是长周期任务还是短周期任务。长周期任务，如前往某地，通常需要多次迭代；而短周期任务，

如拾取物品或与人交谈，迭代次数较少。代理将在接下来的3次交互中遵循新生成的任务。3次交互后，代理返回任务栈中的上一个长周期任务。相比每次迭代重新规划，决定二元任务周期对GPT-4o来说更简单且更稳健。由于

a long-horizon task frequently includes multiple short-horizon sub-tasks, this implementation also helps avoid forgetting the long-horizon tasks under execution.

一个长远任务通常包含多个短期子任务，这种实现方式也有助于避免在执行过程中遗忘长远任务。

**Skill Curation.** As shown in Figure 18, during gameplay, instructions often appear on the screen, such as "press [Q] to take over" and "hold [TAB] to view your stored weapons", which serve as essential directives for completing current and future tasks proficiently. To save interactions with GPT-4o, we implement a simple version of this module inside Information Gathering to reduce interactions with GPT-4o. When GPT-4o detects and classifies some instructional text in the recent observation, which usually contains key and button hints, it will directly generate the corresponding code and description.

技能策划。如图18所示，在游戏过程中，屏幕上经常会出现指令，例如“按 [Q] 接管”和“按住 [TAB] 查看已存储武器”，这些都是完成当前及未来任务的关键指示。为了减少与GPT-4o的交互，我们在信息收集模块内实现了该模块的简化版本。当GPT-4o检测并分类最近观察到的指令文本（通常包含关键字和按键提示）时，会直接生成相应的代码和描述。

**Action Planning.** Upon execution of this module, we first retrieve the top  $k$  relevant skills for the task from procedural memory, alongside the newly generated skills. We then provide GPT-4o with the current task, the set of retrieved skills, and other information collected in Information Gathering that may be helpful for decision-making (e.g., recent screenshots with corresponding descriptions, previous decisions, and examples) and let it suggest which skills should be executed. We also request that GPT-4o provide the reasons for choosing these skills, which increases the accuracy, stability, and explainability of skill selection and thus greatly improves framework performance. While GPT-4o sometimes may generate a sequence of actions, we currently only execute the first one, and perform Self-Reflection, since we observe a tendency for the second action to usually suffer from severe hallucinations.

行动规划。执行此模块时，我们首先从程序记忆中检索与任务最相关的前 $k$ 项技能，以及新生成的技能。然后，我们将当前任务、检索到的技能集合以及在信息收集阶段获得的其他可能有助于决策的信息（例如，最近的截图及其对应描述、先前的决策和示例）提供给GPT-4o，让其建议应执行哪些技能。我们还要求GPT-4o说明选择这些技能的理由，这提高了技能选择的准确性、稳定性和可解释性，从而大幅提升了框架性能。虽然GPT-4o有时会生成一系列动作，但我们目前只执行第一个动作，并进行自我反思，因为我们观察到第二个动作通常会出现严重的幻觉问题。

**Action Execution.** Unlike the conventional mouse operation in standard software, where the cursor is restricted to a 2D grid and remains visible on the screen to navigate and interact with elements, the utilization of the mouse in 3D games like RDR2 introduces a varied control scheme. In menu screens, the mouse behaves traditionally, offering familiar point-and-click functionality. However, during gameplay, the mouse cursor disappears, requiring players to move the mouse according to specific action semantics. For example, to alter the character's viewpoint, the player needs to map the actual mouse movement to in-game direction angle changes, which differ in magnitude in the X and Y axes. Another special transition applies to shooting mode, where the front sight is fixed at the center of the screen, and players must maneuver the mouse to align the sight with target enemies. This nuanced approach to mouse control in different contexts adds an extra layer of challenge to general computer handling, showcasing the adaptability required in game environments, compared to regular software applications.

动作执行。与标准软件中传统的鼠标操作不同，传统软件中光标被限制在二维网格内并始终可见以便导航和交互，而在像《荒野大镖客2》（RDR2）这样的3D游戏中，鼠标的使用引入了多样化的控制方案。在菜单界面中，鼠标表现传统，提供熟悉的点选功能。然而，在游戏过程中，鼠标光标消失，玩家需根据特定的动作语义移动鼠标。例如，为了改变角色的视角，玩家需要将实际鼠标移动映射到游戏内视角角度的变化，且X轴和Y轴的变化幅度不同。另一个特殊转换发生在射击模式中，准星固定在屏幕中心，玩家必须操控鼠标使准星与目标敌人对齐。这种在不同场景下对鼠标控制的细致处理，为一般的电脑操作增加了额外的挑战，展示了游戏环境中相较于常规软件应用所需的适应能力。

Procedural Memory. In our target setting, We intend to let the agent learn all skills from scratch, to the extent possible for the main storyline missions. The procedural memory is initialized with only preliminary skills for basic movement, which are not clearly provided by the in-game tutorial and guidance.

程序性记忆。在我们的目标环境中，我们打算让代理从零开始学习所有技能，尽可能覆盖主线任务。程序性记忆仅初始化了基本移动的初步技能，这些技能并未在游戏内教程和指导下明确提供。

- turn(degree), move\_forward(duration): Since the game does not precisely introduce how to move in the world through in-game instructions, we provide these two basic actions in advance, so GPT-4o can perform basic mobility, while greatly reducing the number of calls to the model.
- 转向（度数），前进（持续时间）：由于游戏中没有通过游戏内指令精确介绍如何在世界中移动，我们提前提供这两个基本动作，以便GPT-4o能够执行基本的移动操作，同时大幅减少对模型的调用次数。
- shoot(x, y): RDR2 also does not provide detailed instructions on how to aim and shoot. Moreover, due to limitations with GPT-4o spatial reasoning and the need to sometimes augment images with object bounding boxes, we provide such basic skill for the agent to complete relevant tasks.
- 射击(x, y): RDR2（荒野大镖客2）同样未提供关于如何瞄准和射击的详细指导。此外，由于GPT-4o在空间推理方面的限制，以及有时需要通过物体边界框增强图像，我们为代理提供了此类基本技能以完成相关任务。
- select\_item\_at (x, y) : Similarly to shoot(), due to the lack of instructions, we provide such skill for the agent to move the mouse to a certain place to select a given item.
- select\_item\_at (x, y) : 与shoot()类似，由于缺乏具体指令，我们为智能体提供此技能，以便将鼠标移动到指定位置选择给定的物品。

Beyond these basic atomic low-level actions, we introduce a few composite skills to facilitate the game playing progress. The agent should be able to complete tasks using only the basic skills above and the skills it learns, but these composite skills streamline the process by greatly reducing calls to the backend model.

除了这些基本的原子级低层动作外，我们引入了一些复合技能以促进游戏进程。代理应能够仅使用上述基本技能和其学习到的技能完成任务，但这些复合技能通过大幅减少对后端模型的调用，使过程更加简化。

- turn\_and\_move\_forward(degree, duration): This skill is just a simple composition of turn() and move\_forward() to save frequent calls to GPT-4o in a common sequence.
- turn\_and\_move\_forward(角度, 持续时间): 此技能只是turn()和move\_forward()的简单组合，用于节省在常见序列中频繁调用GPT-4o的次数。
- follow(duration) and navigate\_path(duration): In RDR2, tasks often guide players to follow NPCs or generated paths (red lines) in the minimap to certain locations. This can be reliably
- follow(duration) 和 navigate\_path(duration): 在《荒野大镖客救赎2》（RDR2）中，任务通常引导玩家在小地图上跟随NPC或生成的路径（红线）前往特定地点。这一过程可以被可靠地

accomplished via the basic movement skills, but requires numerous interactions with GPT-4o. To control both cost and time budgets involving GPT-4o's responses, we leverage the information shown in the minimap to implement a composite skill to follow target NPCs or red lines for a short set of game iterations. The default duration is 20 iterations. Increasing the duration can dramatically improve the performance in task Follow Dutch, Follow Javier and Killing Wolves but significantly decrease the success rate of Search John since this task requires frequent exchange of the skills between climbing and following.

通过基本的移动技能实现，但需要与GPT-4o进行大量交互。为了控制涉及GPT-4o响应的成本和时间预算，我们利用小地图中显示的信息，实施一种复合技能，在短时间的游戏迭代中跟随目标NPC或红线。默认持续时间为20次迭代。延长持续时间可以显著提升“跟随Dutch”、“跟随Javier”和“猎杀狼”任务的表现，但会显著降低“搜索John”任务的成功率，因为该任务需要频繁在攀爬和跟随技能之间切换。

- fight(): As output of an interaction with GPT-4o, the agent will only take one action per step. However, though the action is generated correctly, specifically in fight scenarios, the action frequency may not be high

enough to defeat an opponent. In order to allow sub-second punches, we provide a pre-defined action that wraps this multi-action punching, which can be selected by GPT-4o to effectively win fights.

- `fight()`: 作为与GPT-4o交互的输出，代理每步只执行一个动作。然而，尽管动作生成正确，特别是在战斗场景中，动作频率可能不足以击败对手。为了实现亚秒级的连击，我们提供了一个预定义动作，封装了多次连击动作，GPT-4o可以选择该动作以有效赢得战斗。

For the open-ended mission, since the agent skips all the tutorials in Chapter I, we provide all the necessary skills in the procedural memory at the beginning of the mission.

对于开放式任务，由于代理跳过了第一章的所有教程，我们在任务开始时将所有必要技能存储在程序记忆中提供给代理。

Episodic Memory. This module stores all the useful information, e.g., input and output of GPT-4o. In each iteration, after the self-reflection, we will request GPT-4o to summary the event that happened in the last action and the past experiences.

情景记忆（Episodic Memory）。该模块存储所有有用信息，例如GPT-4o的输入和输出。在每次迭代中，自我反思后，我们会请求GPT-4o总结上一次动作发生的事件及过去的经验。

Game Pause. To prevent in-game time from passing in real-time games like RDR2, we have to pause the game while waiting for LMMs' response. The time interval between two consecutive actions can be as long as one minute. In RDR2, after the agent finishes executing outputted actions, esc will be automatically pressed to pause the game and when the agent determines the next action, esc will be automatically pressed again to unpause the game. Note that there will be an animation lasting up to 0.5 seconds for both pausing and unpausing. During this animation, we can not control the character, but the dynamics of the game world keep changing, e.g., the wolves are still moving. It introduces additional challenges for the tasks that require precise timing, like combat.

游戏暂停。为了防止像《荒野大镖客2》（RDR2）这类实时游戏中的游戏时间流逝，我们必须在等待大语言模型（LMMs）响应时暂停游戏。两次连续动作之间的时间间隔最长可达一分钟。在RDR2中，代理执行完输出动作后，系统会自动按下Esc键暂停游戏；当代理确定下一动作时，Esc键会再次自动按下以取消暂停。注意，暂停和取消暂停时各有最长0.5秒的动画，在此期间无法控制角色，但游戏世界的动态仍在变化，例如狼群仍在移动。这为需要精确时机的任务（如战斗）带来了额外挑战。

## 26.7 B.4 Case Studies

### 26.8 B.4 案例研究

Here we present a few game-specific case studies for more in-depth discussion of the framework capabilities and the challenges of the GCC setting.

这里我们展示几个针对特定游戏的案例研究，以更深入讨论框架能力及GCC设置的挑战。

#### 26.8.1 B.4.1 Self-Reflection

##### 26.8.2 B.4.1 自我反思

Self-reflection is an essential component in CRADLE as it allows our framework reasoning to correct previous mistakes or address ineffective actions taken in-game. Figure 17 provides an example of the self-reflection module. The task requires the agent to select a weapon to equip, in the context of the "Protect Dutch" task. Initially, the agent selects a knife as its weapon by chance, but since the game requires a gun to be chosen, this is incorrect and the game still prompts the player to re-open the weapon wheel. The self-reflection module is able to determine that the previous action was incorrect and on a subsequent iteration the agent successfully opts for the gun, correctly fulfilling the task requirement and advancing to the next stage in the story.

自我反思是CRADLE中的核心组成部分，它使框架能够推理纠正之前的错误或解决游戏中采取的无效动作。图17展示了自我反思模块的示例。任务要求代理在“保护Dutch”任务背景下选择装备武器。起初，代理偶然选择了刀作为武器，但游戏要求选择枪械，因此这是错误的，游戏仍提示玩家重新打开武器轮盘。自我反思模块能够判断之前的动作错误，在后续迭代中代理成功选择了枪械，正确完成任务要求并推进故事下一阶段。

### **26.8.3 B.4.2 Skill Curation**

#### **26.8.4 B.4.2 技能策划**

For skill curation, we first provide GPT-4o with examples of general mouse and keyboard control APIs, e.g., `io_env.key_press` and `io_env.mouse_click`. Figure 18 shows that GPT-4o can capture and understand the prompts appearing on screenshots, i.e., icons and text, and strictly follow the provided skill examples using our IO interface to generate correct skill code. Moreover, GPT-4o also generates comments in the code to demonstrate the functionality of this skill, which are essential for computing similarity and relevance with a given task during skill retrieval. The quality of the generated comment directly determines the results of skill retrieval, and further impacts reasoning to action planning. Curation can also re-generate code for a given skill, which is useful if GPT-4o wrongly recognized a key or mouse button in a previous iteration.

在技能策划中，我们首先向GPT-4o提供了通用鼠标和键盘控制API示例，如`io_env.key_press`和`io_env.mouse_click`。图18显示GPT-4o能够捕捉并理解截图中出现的提示（图标和文本），并严格按照提供的技能示例，使用我们的IO接口生成正确的技能代码。此外，GPT-4o还会在代码中生成注释，说明该技能的功能，这对于计算与给定任务的相似度和相关性以进行技能检索至关重要。生成注释的质量直接决定技能检索结果，进而影响推理到动作规划。策划还可以为给定技能重新生成代码，这在GPT-4o之前迭代错误识别键或鼠标按钮时非常有用。

#### **26.8.5 B.4.3 Action Execution and Feedback**

#### **26.8.6 B.4.3 动作执行与反馈**

Proper reasoning about environment feedback is critical due to the generality of the GCC setting and the level of abstraction to interact with the complex game world. The semantic gaps between

由于GCC设置的通用性及与复杂游戏世界交互的抽象层次，正确推理环境反馈至关重要。动作执行、其在游戏世界中的效果以及观察相关结果以进行进一步推理之间的语义差距导致了CRADLE需要处理的若干潜在问题。

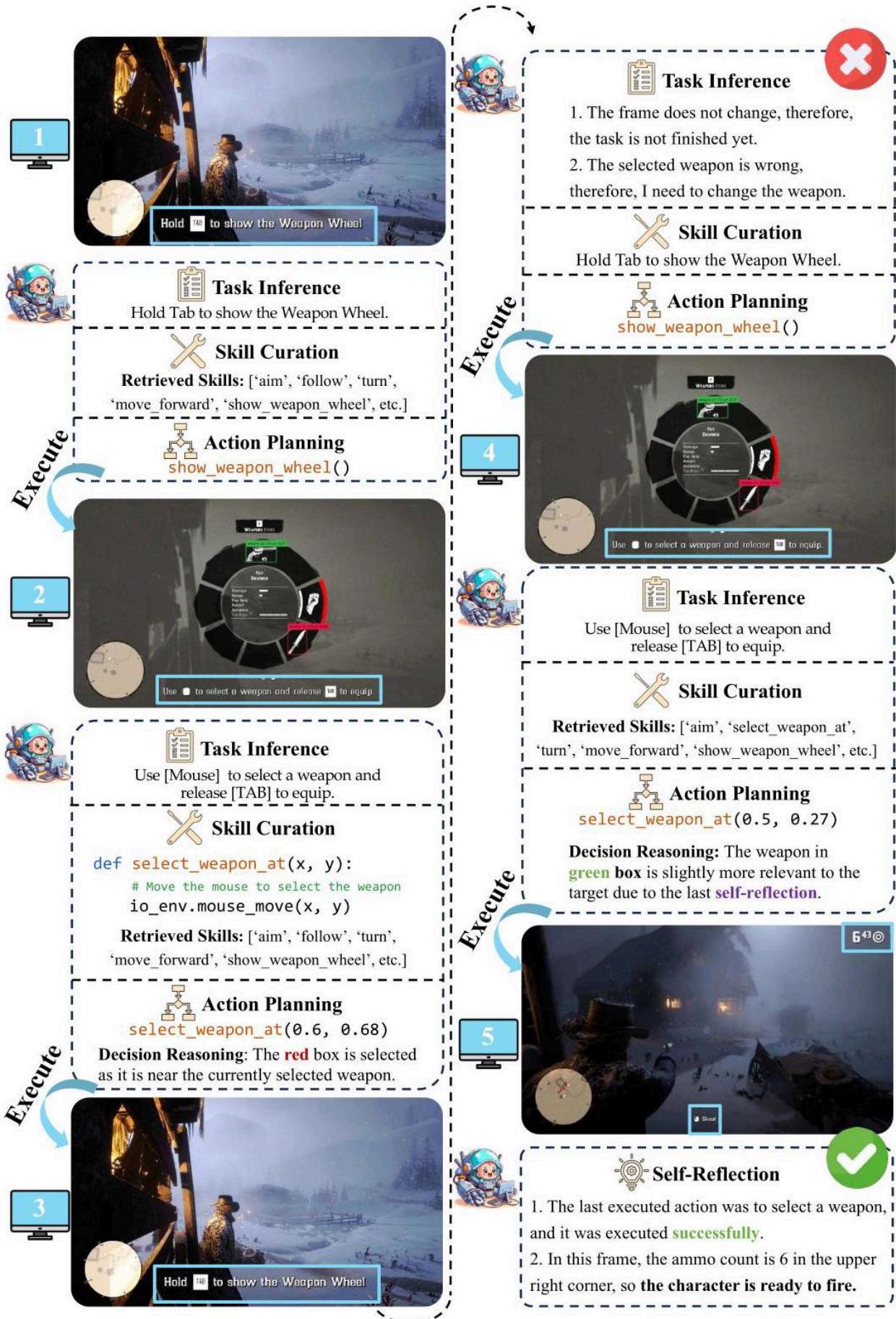


Figure 17: Case study of self-reflection on re-trying a failed task. Task instruction and context require the agent to equip the gun. A wrong weapon (knife) is first selected, but the agent equips the gun after self-reflection. Only relevant modules are shown for better readability, though all modules (Figure 4) are executed per iteration.

图17：自我反思重试失败任务的案例研究。任务指令和背景要求代理装备枪械。起初错误选择了刀，但经过自我反思后代理装备了枪。为提高可读性，仅显示相关模块，尽管每次迭代都会执行所有模块（见图4）。



Figure 18: Skill code generation based on in-game instructions. As the storyline progresses, the game will continually provide prompts on how to use a new skill via keystrokes or utilizing the mouse.

图18：基于游戏内指令的技能代码生成。随着剧情推进，游戏会持续通过按键或鼠标操作提示如何使用新技能。

the execution of an action, its effects in the game world, and observing the relevant outcomes for further reasoning lead to several potential issues that CRADLE needs to deal with. Such issues can be categorized into four major cases:

动作执行、其在游戏世界中的效果与观察相关结果以进行进一步推理之间的语义差距导致了CRADLE需要处理的若干潜在问题。这些问题可归纳为四大类：

Lack of grounding feedback. In many situations, due to the lack of precise information from the environment, it can be difficult for the system to deduce the applicability or outcome of a given action. For example, when picking an item from the floor, the action may fail due to the distance to the object not yet being close enough. Or, if within pick up range, the chosen action may not exactly apply due to other factors (e.g., character's package is full).

缺乏基础反馈。在许多情况下，由于环境缺乏精确信息，系统难以推断某动作的适用性或结果。例如，从地上拾取物品时，动作可能因与物体距离尚未足够近而失败；或者即使在拾取范围内，所选动作也可能因其他因素（如角色包裹已满）而无法执行。

Even if the right action is selected and executed successfully, the agent still needs to figure out its results from the partial visual observation of the game world. If the agent needs to pick or manipulate an object that is occluded from view, the action may execute correctly, but no outcome can be seen.

即使选择并成功执行了正确的动作，智能体仍需从游戏世界的部分视觉观察中推断其结果。如果智能体需要拾取或操作一个被遮挡的物体，动作可能执行正确，但无法看到任何结果。

A representative example in RDR2 happens when the agent tries to pick up its gun from the floor after a fight. Getting to the right distance, without completely occluding the object, can lead to multiple re-trials. Figure 19a showcases a situation where, though the character is already standing near the gun (as seen in the minimap), it's still not possible to pick it up.

在《荒野大镖客2》（RDR2）中的一个典型例子是，智能体在战斗后尝试从地上捡起枪。达到合适距离且不完全遮挡物体，可能导致多次重试。图19a展示了这样一种情况：尽管角色已经站在枪附近（如小地图所示），仍无法捡起枪。

Previous efforts [57, 61] that utilize in-game state APIs unreasonably bypass such issues by leveraging internal structured information from the game and the full semantics of responses (data) or failures (error messages). 之前的研究[57, 61]利用游戏内状态API，不合理地绕过了这些问题，通过利用游戏的内部结构化信息以及响应（数据）或失败（错误信息）的完整语义。

Imprecise timing in IO-level calls. This issue is caused by the ambiguity in the game instructions or differences in specific in-game action behaviors, where even the execution of a correct action may fail due to minor timing mismatches. For example, when executing an action like 'open cabinet', which requires pressing the [R] key on the keyboard, if the press is too fast, no effect happens in the game world. However, as there is no visual change in the game nor other forms of feedback, it can be difficult for GPT-4o to figure out if an inappropriate action was chosen at this game state or if the minor timing factor was the problem. Pressing the key for longer triggers an animation around the button (only if the helper menu is on screen), but this is easily missed and any key release before the circle completes also results in no effect. Figure 19b illustrates the situation.

输入输出层调用时序不精确。该问题源于游戏指令的模糊性或特定游戏操作行为的差异，即使执行了正确动作，也可能因微小的时序不匹配而失败。例如，执行“打开柜子”动作时，需要按下键盘上的[R]键，如果按得过快，游戏中不会产生效果。然而，由于游戏中没有视觉变化或其他反馈，GPT-4o难以判断是在该游戏状态下选择了不合适的操作，还是时序因素导致的问题。长按按键会触发按钮周围的动画（仅当辅助菜单显示时），但这很容易被忽略，且在动画完成前松开按键同样无效。图19b说明了这一情况。

The same problem also manifests in other situations in the game, where pressing the same key for longer triggers a completely different action (e.g., lightly pressing the [Left Alt] key vs. holding it for longer).

同样的问题也出现在游戏中的其他情境中，长按同一按键会触发完全不同的动作（例如，轻按[左Alt]键与长按的区别）。

Change in the semantics of key and button. A somewhat similar situation occurs when the same keyboard key or mouse button gets attributed different semantics in different situations (or even in a multi-step action). GPT-4o may decide to execute a given skill, but the original semantics no longer hold. The lack of in-game effect parallels the previous situations. Worse yet, an undesired effect will confuse the system regarding the correct action being selected or not.

按键和按钮语义的变化。类似情况发生在同一键盘键或鼠标按钮在不同情境（甚至多步骤动作中）被赋予不同语义时。GPT-4o可能决定执行某项技能，但原有语义不再适用。缺乏游戏内效果与之前情况类似。更糟的是，产生的非预期效果会使系统混淆是否选择了正确动作。

For example, when approaching a farm in the beginning of the game, the agent needs to hitch the horse to a pole to continue. The operation to perform the action consists of pressing the [E] key near a hitching post (as shown in Figure 19c). However, the same [E] key press is the only constituting step in other actions with different semantics, like dismount the horse or open the door. Wrongly triggering a horse dismount at the situation shown in the figure can lead to undesired side effects, i.e.,

例如，在游戏初期接近农场时，智能体需要将马拴在柱子上以继续。执行该动作需在拴马桩附近按下[E]键（如图19c所示）。然而，同一[E]键在其他动作中也作为唯一步骤，且语义不同，如下马或开门。在图示情境中错误触发下马动作可能导致不良副作用，即



(a) 'Pick gun' unavailable

(b) 'Open cabinet' press timing

(c) 'Hitch horse' re-use of [E] key

Figure 19: Examples of action execution uncertainty. Lack of environmental feedback to actions and semantic gaps between action intent and game command can lead to challenging situations for agent reasoning.

图19：动作执行不确定性的示例。动作缺乏环境反馈以及动作意图与游戏指令之间的语义差距，可能导致智能体推理面临挑战。

it may mislead the system about the actual effects of the action or affect the planning of which next actions to perform.

这可能误导系统对动作实际效果的判断，或影响后续动作的规划。

Interference issues. Lastly, completion of some actions requires the correct execution of multiple steps sequentially, which could be interrupted in many ways not related to the agent's own actions. Without the use of APIs that expose internal states or other forms of feedback, it is much harder for the agent to decide when to repeat sub-actions or try different strategies. For example, if the agents gets shot and loses aim while in combat, or an unrelated in-game animation is triggered mid-action, canceling it.

干扰问题。最后，某些动作的完成需要正确顺序执行多个步骤，这些步骤可能被许多与智能体自身动作无关的因素中断。若不使用暴露内部状态或其他反馈形式的API，智能体更难判断何时重复子动作或尝试不同策略。例如，智能体在战斗中被射击失去瞄准，或在动作中途触发无关动画导致动作取消。

Since there is no direct environment feedback, the agent needs to carefully analyze the situation and try to infer if any action step needs re-execution.

由于没有直接的环境反馈，智能体需要仔细分析情况，尝试推断是否需要重新执行某个动作步骤。

## 26.9 B.5 Limitations of GPT-4o and GPT-4V

### 26.10 B.5 GPT-4o和GPT-4V的局限性

Deploying CRADLE in a complex game like RDR2 requires the backbone LMM model to handle multimodal input, which revealed several limitations of both GPT-4V and GPT-4o, necessitating external tools to enhance overall framework performance. Initial tests and exploration were performed using GPT-4V, as GPT-4o was not yet available. These tests highlighted significant weaknesses in spatial perception, icon understanding, history processing, and world understanding. Upon the release of GPT-4o, further testing demonstrated some notable improvements in spatial perception. However, enhancements in other areas remained marginal, while some regressions were also observed, all indicating the need for additional tools to aid decision-making.

在像《荒野大镖客2》这样复杂的游戏中部署CRADLE，需要骨干大规模语言模型（LMM）处理多模态输入，这暴露了GPT-4V和GPT-4o的若干局限，需借助外部工具提升整体框架性能。最初测试和探索使用了GPT-4V，因为当时GPT-4o尚未发布。测试揭示了空间感知、图标理解、历史处理和世界理解方面的显著弱点。GPT-4o发布后，进一步测试显示空间感知有所改善，但其他方面提升有限，且出现了一些退步，均表明需要额外工具辅助决策。

Spatial Perception. As shown in Figure 20a and 21a, GPT-4V's spatial-visual recognition capability is insufficient for precise fine-grained control, particularly in detecting whether the character is being or going to be blocked and in estimating the accurate relative positions of target objects. In contrast, GPT-4o exhibits a significant enhancement in spatial perception, capable of recognizing obstacles ahead and estimating the approximate relative positions between objects. However, both models require supplementary information, such as bounding boxes of potential target objects, to make fine-grained decisions. These led to the need to augment certain images to provide auxiliary visual clues for decision-making, i.e., bounding boxes of possible target objects.

空间感知。如图20a和21a所示，GPT-4V的空间视觉识别能力不足以实现精细的细粒度控制，尤其是在检测角色是否被阻挡或即将被阻挡以及估算目标物体的准确相对位置方面。相比之下，GPT-4o在空间感知方面有显著提升，能够识别前方障碍物并估算物体间的大致相对位置。然而，两种模型都需要辅助信息，如潜在目标物体的边界框，才能做出细粒度决策。因此，需要对某些图像进行增强，以提供辅助视觉线索用于决策，即可能目标物体的边界框。

Icon Understanding. Both GPT-4o and GPT-4V struggle with domain-specific concepts, such as unique icons within the game, which may represent specific targets or refer to certain mouse and key actions. As shown in Figure 20b and 21b, GPT-4V and GPT-4o fail to recognize the left shift, right mouse button, and space icons. Attempts to incorporate few-shot learning to improve image understanding cannot be generalized. Therefore, we match prepared pattern templates, e.g., icon images, against each screenshot to continuously detect and highlight any appearing icons.

图标理解。GPT-4o和GPT-4V都难以处理领域特定概念，例如游戏中的独特图标，这些图标可能代表特定目标或指代某些鼠标和按键操作。如图20b和21b所示，GPT-4V和GPT-4o未能识别左Shift键、鼠标右键和空格键图标。尝试通过少量示例学习提升图像理解能力未能实现泛化。因此，我们通过匹配预先准备的模式模板（如图标图像）与每个截图，持续检测并高亮显示出现的任何图标。

Figure 20c and Figure 21c also demonstrate that although GPT-4o performs better than GPT-4V in understanding the mini-map, it still fails to consistently interpret this crucial information regarding the position and direction of the character. This failure in localization leads the agent to sometimes get lost in the town and miss the task target. While the aforementioned issues can be slightly alleviated by providing additional few-shot examples, a significant improvement is only achieved by cropping the image and providing GPT-4o with the exact region containing the icon to be recognized. This dependency on precise input makes the issue challenging and unreliable for decision-making. Although the above issues can be slightly alleviated by providing additional few-shot examples, it

图20c和图21c还显示，尽管GPT-4o在理解小地图方面优于GPT-4V，但仍未能持续准确解读关于角色位置和方向的关键信息。这种定位失败导致智能体有时在城镇中迷路，错过任务目标。上述问题虽可通过提供额外的少量示例略微缓解，但只有通过裁剪图像并向GPT-4o提供包含待识别图标的精确区域，才能实现显著改进。这种对精确输入的依赖使得问题在决策时具有挑战性且不可靠。尽管上述问题可通过提供额外少量示例略微缓解，



 ChatGPT

There are **no immediate obstacles** directly in front of the character in the image. The character has a **clear path ahead** on a muddy street in what appears to be a Western-style town setting.



 ChatGPT

The shopkeeper in the image is on the **right side**, behind the counter.

- (a) GPT-4V fails to recognize obstacles in the environment and the position of the shopkeeper.
- (a) GPT-4V未能识别环境中的障碍物及店主的位置。



### ChatGPT

Based on the provided game screenshot, I've identified the following control prompt:

- "Hold L1 to ride faster"

This control suggests that holding a certain key (which seems to be an **upward arrow key**, often associated with the '**W**' key for forward movement in many games) will make the character ride faster.



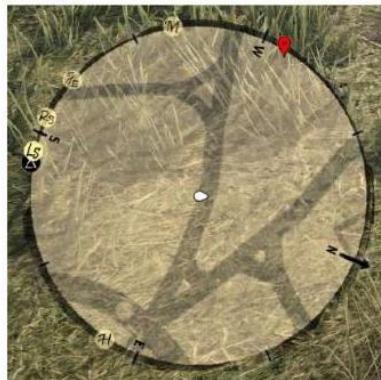
### ChatGPT

From the provided image screenshot from the game, I can identify the following keyboard and mouse control prompts:

1. "**Toggle View**" - The key for this action is "**V**".
2. "**Zoom**" - The key for this action is "**O**".
3. "**Flip**" - The key for this action is "**L**".
4. "**Back**" - The key for this action is "**ESC**".

(b) GPT-4V struggles to recognize the icons for keys on keyboard and mouse buttons.

(b) GPT-4V难以识别键盘和鼠标按钮的图标。



### ChatGPT

The central white arrow is facing towards the **left**.



### ChatGPT

The central white arrow with the white arrowhead is facing towards the **left**.

(c) GPT-4V cannot understand the correct direction of arrow points, i.e., character, towards in the mini-map.

Figure 20: Example situations of GPT-4V's limitations in understanding visual information from the game.

(c) GPT-4V无法理解小地图中箭头指向的正确方向，即角色所朝向的方向。图20：GPT-4V在理解游戏视觉信息方面的局限性示例情境。



## 27 ChatGPT

## 28 ChatGPT

ChatGPT

ChatGPT

There is an obstacle in front of the character. It appears to be a small wooden railing or barrier along the edge of the platform. The character would need to navigate around it or find a way over it to proceed forward. The shopkeeper in the image is on the left side

角色前方有一个障碍物，看起来像是平台边缘的一小段木质栏杆或障碍。角色需要绕过它或找到越过它的方法才能继续前进。图中的店主位于左侧。

(a) GPT-4 can recognize obstacles in the environment and the position of the shopkeeper.

(a) GPT-4o能够识别环境中的障碍物及店主的位置。



## 29 ChatGPT

## 30 ChatGPT

## 31 ChatGPT

## 32 ChatGPT

Based on the provided image, the control prompt shown is: From the provided image screenshot from the game, 根据提供的图像，显示的控制提示为：从游戏提供的截图中，

- "Hold L to ride faster".
- “按住L以加速骑行”。

I can identify the following keyboard and mouse control prompts:

我可以识别以下键盘和鼠标控制提示：

The 'L' refers to the left analog stick on a controller, indicating that the player should hold the left analog stick in a forward position to increase the speed of their character's movement, typically while riding a horse. 1. Toggle View: V

“L”指的是手柄上的左摇杆，表示玩家应将左摇杆向前推以增加角色的移动速度，通常是在骑马时。1. 切换视角：V

2. Zoom: Mouse Scroll Wheel

3. 缩放：鼠标滚轮

3. Flip: R

4. 翻转：R键

4. Back: Esc

5. 返回：Esc键

5. Photograph: (No specific key assigned, just a

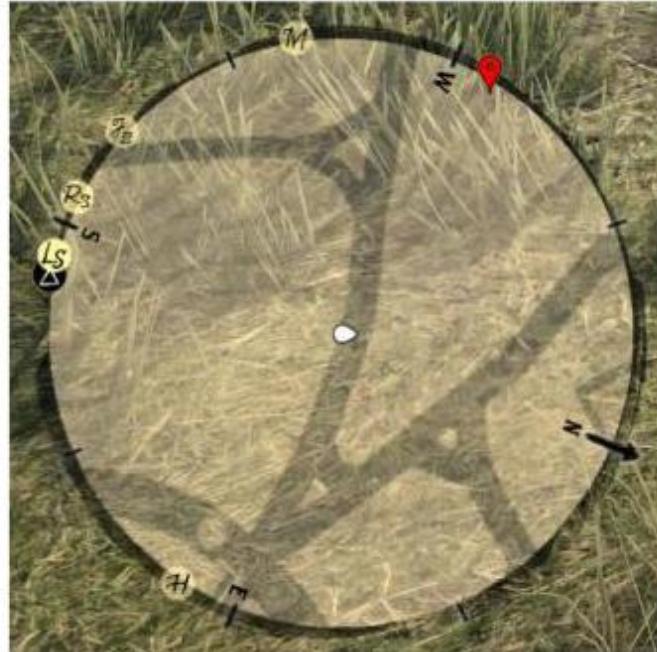
6. 拍照: (未指定具体按键, 只是一个

label)

标签)

(b) GPT-4o struggles to recognize the icons for keys on keyboard and mouse buttons.

(b) GPT-4o难以识别键盘按键和鼠标按钮的图标。



### 33 ChatGPT

### 34 ChatGPT

### 35 ChatGPT

### 36 ChatGPT

The central white arrow in the mini-map is facing  
小地图中央的白色箭头朝向

to the right. down.

右侧。下方。

(c) GPT-4o occasionally accurately determines the direction of arrow points, i.e., character, towards in the mini-map.

(c) GPT-4o偶尔能准确判断小地图中箭头指向的方向, 即角色朝向。

Figure 21: Example situations of GPT-4o's limitations in understanding visual information from the game.

图21: GPT-4o在理解游戏视觉信息方面局限性的示例情境。

can only have an obvious effect if we crop the image and provide the GPT-4o with the region exactly containing the icon to be recognized, which makes the issue intractable.

只有当我们裁剪图像并向GPT-4o提供恰好包含待识别图标的区域时, 才能产生明显效果, 这使得问题难以解决。

History Processing. Moreover, both GPT-4o and GPT-4V can easily get distracted by irrelevant information in longer contexts, resulting in hallucinations. For example, when action planning utilizes too many historical screenshots, they may confuse past and present frames. Additionally, performance fluctuates and both model versions frequently generate output not adhering to the rules in the provided prompts. To mitigate the issue of hallucinations, we more strictly control input information by further summarizing long-term memory.

历史处理。此外，GPT-4o和GPT-4V在较长上下文中容易被无关信息干扰，导致幻觉现象。例如，当动作规划使用过多历史截图时，模型可能混淆过去与当前帧。此外，性能波动较大，两个模型版本经常生成不符合提示规则的输出。为缓解幻觉问题，我们通过进一步总结长期记忆，更严格地控制输入信息。

World Understanding. Lastly, the absence of an RDR2 world model limits GPT-4V and GPT-4o's understanding of the consequences of its actions in the game. This often results in inappropriate action selection, such as overestimating the necessary adjustments for aligning targets or misjudging the duration required for certain actions. To alleviate this problem, we introduced extra prompt rules regarding action parameters and more flexibility into the self-reflection module.

世界理解。最后，缺乏《荒野大镖客救赎2》（RDR2）世界模型限制了GPT-4V和GPT-4o对游戏中行为后果的理解。这常导致动作选择不当，如高估调整目标所需的幅度或误判某些动作所需的时间。为缓解此问题，我们引入了关于动作参数的额外提示规则，并在自我反思模块中增加了更多灵活性。

## 37 C Stardew Valley

### 38 C 星露谷物语

#### 38.1 C.1 Introduction to Stardew Valley

#### 38.2 C.1 星露谷物语简介

Stardew Valley is an open-ended country-life RPG game developed by ConcernedApe, which has a 98% positive rating on Steam and is rated as Overwhelmingly Positive. Players take on the role of a character disillusioned with city life who inherits a dilapidated farm from their late grandfather. Initially, the farmland is overrun with boulders, trees, stumps, and weeds, which players must clear to make way for crops, buildings, and placeable items. The main goal is to restore and expand the farm through activities such as planting crops, raising animals, mining, fishing, and crafting. Additionally, players can interact with NPCs in town, forming relationships that can lead to marriage and children. Players complete quests for money or to restore the town's Community Center by completing "bundles," which reward items like seeds and tools and unlock new areas and game mechanics. All activities are balanced against the character's health, energy, and the game's clock. Food provides buffs, health, and energy. The game features a simplified calendar with four 28-day months representing each season, affecting crop growth and activities. Compared to RDR2, this game is more lightweight and easy to control. This game features a wealth of production and social activities, presenting a comprehensive test of an agent's abilities, which is an ideal platform to observe and evaluate agents' comprehensive behaviors and abilities, like in the Generative Agents [44]. We use the latest version (1.6.8) of the game to conduct all the experiments.

星露谷物语是一款由ConcernedApe开发的开放式乡村生活角色扮演游戏，在Steam上拥有98%的好评率，被评为“极度好评”。玩家扮演一位对城市生活感到失望的角色，继承了已故祖父的一座破旧农场。起初，农田被巨石、树木、树桩和杂草覆盖，玩家必须清理这些障碍，为种植作物、建造建筑和摆放物品腾出空间。主要目标是通过种植作物、养殖动物、采矿、钓鱼和制作工艺品等活动来恢复和扩展农场。此外，玩家可以与镇上的NPC互动，建立关系，最终可能结婚生子。玩家通过完成任务赚取金钱，或通过完成“捆绑包”修复镇上的社区中心，奖励包括种子和工具，并解锁新的区域和游戏机制。所有活动都需平衡角色的健康、能量和游戏时间。食物提供增益效果、恢复健康和能量。游戏采用简化的日历系统，每个季节有四个月，每月28天，影响作物生长和活动。与《荒野大镖客2》（RDR2）相比，本游戏更轻量且易于操作。游戏包含丰富的生产和社交活动，是全面考察智能体能力的理想平台，类似于生成式智能体（Generative Agents）[44]。我们使用游戏的最新版本（1.6.8）进行所有实验。

### 38.3 C.2 Objectives

### 38.4 C.2 目标

We find that GPT-40 surprisingly struggles with accurately recognizing and locating objects near the player in this 2D game. This leads to difficulties for the agent to interact with objects or people, as it requires the player to stand precisely in front of them in the grid (e.g., when entering doors, using a pickaxe to break stones). Even some basic tasks are already challenging enough for current agents in this game. Therefore, as shown in Figure 22, we evaluate three essential tasks in the early stages of the game:

我们发现GPT-40在这款二维游戏中，准确识别和定位玩家附近的物体方面表现令人意外地困难。这导致智能体难以与物体或人物互动，因为它要求玩家必须精确站在网格的正前方（例如，进入门时，使用镐子砸石头时）。即使是一些基本任务，对于当前的智能体来说也已经相当具有挑战性。因此，如图22所示，我们评估游戏早期的三个核心任务：

- Farm Cleanup. Clear the obstacles on the farm, such as weeds, stones, and trees, as much as possible to prepare for farming. This task requires agents to move precisely to be in front of the obstacles, identify the type of obstacles correctly and select corresponding tools to deal with them.
- 农场清理。尽可能清除农场上的障碍物，如杂草、石头和树木，为耕种做准备。该任务要求智能体精确移动至障碍物正前方，正确识别障碍物类型，并选择相应工具进行处理。
- Cultivation. Use the hoe to till the soil, use a parsnip seed packet on the tilled soil to sow a crop, water the crop every day and harvest at least one parsnip. This task requires long-horizontal memory and reasoning.
- 种植。使用锄头耕地，在耕好的土地上使用欧防风（parsnip）种子包播种作物，每天浇水，并至少收获一颗欧防风。该任务需要较长时间的横向记忆和推理能力。
- Shopping. Go to the general store in the town, which is on the other map, to buy more seeds and return home. This task is used to evaluate agents' long-distance navigation ability.
- 购物。前往位于另一张地图上的镇上杂货店购买更多种子，然后返回家中。该任务用于评估智能体的长距离导航能力。

For each task, the maximal steps is 100 .

每个任务的最大步数为100步。



Figure 22: Three tasks in Stardew Valley.

图22：星露谷物语中的三个任务。

### 38.5 C.3 Implementation Details

#### 38.6 C.3 实现细节

**Visual Prompting.** As a cartoon-style pixel game, the game screen of Stardew is quite different from the real world. Although GPT-4o can observe coarse-grained information from screenshots, more fine-grained information is required to complete tasks. Therefore, as shown in Figure 23, we divide each screenshot into  $3 \times 5$  grids and require GPT-4o to describe the screenshot in a grid-by-grid format. We empirically find that it can result in a more precise and accurate description. And GPT-4o can also make better control based on the grids. In addition, we also augment the image with two blue and yellow bands on the left and right sides, respectively, with the prompt, "The blue band represents the left side and the yellow band represents the right side". Our empirical results show that this method significantly improves GPT-4o's ability to accurately distinguish left from right.

**视觉提示。**作为一款卡通风格的像素游戏，星露谷(Stardew)的游戏画面与现实世界有很大不同。虽然GPT-4可以从截图中观察到粗粒度的信息，但完成任务需要更细粒度的信息。因此，如图23所示，我们将每张截图划分为 $3 \times 5$ 个网格，并要求GPT-4以网格为单位描述截图。经验表明，这样可以得到更精确准确的描述，GPT-4也能基于网格做出更好的控制。此外，我们还在图像左右两侧分别添加了蓝色和黄色条带，并附加提示“蓝色条带代表左侧，黄色条带代表右侧”。我们的实验证明，该方法显著提升了GPT-40准确区分左右的能力。

**Information Gathering.** As mentioned in the introduction of visual prompting, we let GPT-4o describe the image grid by grid, which is helpful in locating the position of the character, surrounding objects and buildings and facilitates the understanding of the relative positions among them for GPT-4o. Besides, while compared to GPT-4V, GPT-4o is able to recognize most of the icons and their quality in the toolbar shown at the bottom of the screenshot, GPT-4o cannot output the items in the inventory sequentially one by one as it always skips a few in between. We have to clip the box for each item out of the toolbar and feed them to GPT-4o independently, augmented with template matching, for recognition, which turns out to be more accurate. The success of

recognition of the tools in the toolbar is critical to tasks like Farm Cleanup and Cultivation.

信息收集。如视觉提示介绍中所述，我们让GPT-40逐网格描述图像，这有助于定位角色、周围物体和建筑的位置，并促进GPT-40理解它们之间的相对位置。此外，与GPT-4V相比，GPT-40能够识别截图底部工具栏中的大多数图标及其品质，但无法按顺序逐一输出背包中的物品，因为它总是跳过其中几个。我们不得不将工具栏中每个物品的框单独裁剪出来，独立输入给GPT-40，并辅以模板匹配进行识别，结果更为准确。工具栏中工具的识别成功对于农场清理(Farm Cleanup)和耕作(Cultivation)等任务至关重要。

**Self-Reflection.** The duration of actions in Stardew is usually much shorter than in RDR2, so we only use the first and last frame from the video observation to reduce the number of tokens used per request. Additionally, we provide some helpful prior information for GPT-40. For example, a screenshot of the inside of the store is provided to check whether the store was successfully entered. This is useful because there are many other buildings near the store, and sometimes GPT-40 controls the character to enter the wrong one. However, this is not realized if the screenshot is not provided.

自我反思。星露谷中动作持续时间通常远短于《荒野大镖客2》(RDR2)，因此我们仅使用视频观察的第一帧和最后一帧，以减少每次请求使用的token数量。此外，我们为GPT-40提供了一些有用的先验信息。例如，提供商店内部的截图以检查是否成功进入商店。这很重要，因为商店附近有许多其他建筑，有时GPT-40会控制角色进入错误的建筑，如果不提供截图则无法察觉。

**Skill Curation.** For skill curation, as mentioned in Figure 5, we mainly rely on the in-game manual to generate atomic skills, like move\_up(), do\_action() and use\_tool(). In addition, to handle the challenges of locating objects, especially doors, we have a special set of composite skills specifically for Stardew. e.g., go\_through\_door, buy\_item, get\_out\_of\_house and enter\_door\_and\_sleep. With the restrictions of GPT-40 in fine-grained control, we designed go\_through\_door composite skills for

技能策划。如图5所示，技能策划主要依赖游戏内手册生成原子技能，如move\_up()、do\_action()和use\_tool()。此外，为应对定位物体（尤其是门）的挑战，我们为星露谷专门设计了一套复合技能，例如go\_through\_door、buy\_item、get\_out\_of\_house和enter\_door\_and\_sleep。鉴于GPT-40在细粒度控制上的限制，我们设计了go\_through\_door复合技能用于



Figure 23: Augmented screenshot via visual prompting. The full screenshot is divided into  $3 \times 5$  grids and each grid has a unique white coordinate. Additionally, we augment all input images with color bands, with the prompt, "The blue band represents the left side and the yellow band represents the right side", which significantly improves GPT-4o's ability to accurately distinguish left from right.

图23：通过视觉提示增强的截图。完整截图被划分为 $3 \times 5$ 个网格，每个网格有唯一的白色坐标。此外，我们为所有

输入图像添加了彩色条带，并附加提示“蓝色条带代表左侧，黄色条带代表右侧”，这显著提升了GPT-40准确区分左右的能力。

the agent to control the game character to accurately reach various doors and successfully enter, such as the house and the store door. and in order to buy certain items such as parsnip seeds, we designed the composite skills buy\_item to control the game character to interact with the salesman and buy parsnip seeds. similarly, we designed the get\_out\_of\_house and enter\_door\_and\_sleep composite skills to accurately exit the house from the bed and enter the house and walk to the bed.

代理控制游戏角色准确到达各个门并成功进入，如房屋和商店门。为了购买某些物品如欧防风种子(parsnip seeds)，我们设计了复合技能buy\_item，控制角色与商人互动并购买欧防风种子。同样，我们设计了get\_out\_of\_house和enter\_door\_and\_sleep复合技能，实现从床上准确离开房屋，以及进入房屋并走向床铺。

Action Planning. In this game, we let GPT-4o output at most two skills in a single action every time, which turns out to be efficient. The agent usually needs to select the correct tool first and then use the tool or do action.

动作规划。在本游戏中，我们让GPT-40每次输出最多两个技能的单一动作，事实证明这样效率较高。代理通常需要先选择正确的工具，然后使用该工具或执行动作。

Procedure Memory. Procedure Memory is used to store and retrieve skills in code form. In order for agents to quickly get started and complete some special tasks in Stardew, we have predefined skills in Procedure Memory. These skills are divided into atomic and composite skills. atomic skill consists of basic operations such as moving, selecting tools, etc. The description of all the atomic skills is listed as follows:

过程记忆。过程记忆用于存储和检索代码形式的技能。为了让代理快速启动并完成星露谷中的一些特殊任务，我们在过程记忆中预定义了技能。这些技能分为原子技能和复合技能。原子技能包括移动、选择工具等基本操作。所有原子技能的描述如下：

- do\_action(): The function to perform a context-specific action on objects or characters.
- do\_action(): 对对象或角色执行上下文相关动作的函数。
- use\_tool(): The function to execute an in-game action commonly assigned to using the character's current selected tool.
- use\_tool(): 执行游戏中通常分配给使用当前选中工具的动作的函数。
- move\_up(duration): The function to move the character upward (south) by pressing the 'w' key for the specified duration.
- move\_up(duration): 通过按下“w”键指定时间，使角色向上（南）移动的函数。
- move\_down(duration): The function to move the character downward (north) by pressing the 'w' key for the specified duration.
- move\_down(duration): 通过按下“w”键指定时间，使角色向下（北）移动的函数。
- move\_left(duration): The function to move the character left (west) by pressing the 'w' key for the specified duration.
- move\_left(duration): 通过按下“w”键指定时间，使角色向左（西）移动的函数。
- move\_right(duration): The function to move the character right (east) by pressing the 'w' key for the specified duration.
- move\_right(duration): 通过按下“w”键指定时间，使角色向右（东）移动的函数。
- select\_tool(key): The function to select a specific tool from the in-game toolbar based on the given tool number.
- select\_tool(key): 根据给定的工具编号，从游戏内工具栏中选择特定工具的函数。

and the composite skills are designed for the agent to complete a variety of special tasks. The description of all the composite skills is listed as follows:

复合技能旨在让智能体完成各种特殊任务。所有复合技能的描述如下：

- buy\_item(): The function to interact with the salesman and buy the item.  
• buy\_item(): 与商人交互并购买物品的函数。
- enter\_door\_and\_sleep(): The function to enter the house and walk to the bed.  
• enter\_door\_and\_sleep(): 进入房屋并走向床铺的函数。
- get\_out\_of\_house(): The function to accurately exit the house from the bed  
• get\_out\_of\_house(): 从床铺准确地走出房屋的函数。
- go\_through\_door(): The function to reach and enter all kinds of doors.  
• go\_through\_door(): 到达并进入各种门的函数。

Game Pause. The game will pause automatically when the game window is not focused. So when the character finishes executing actions, we will activate another window, e.g., code window, to pause the game and stop the passage of the time in the game.

游戏暂停。当游戏窗口未获得焦点时，游戏会自动暂停。因此，当角色完成动作后，我们会激活另一个窗口，例如代码窗口，以暂停游戏并停止游戏内时间流逝。

### 38.7 C.4 Case Studies

### 38.8 C.4 案例研究

Here we present a few game-specific case studies to further discuss CRADLE's self-reflection and task-inference processes in the GCC setting.

这里展示几个特定游戏的案例研究，进一步讨论CRADLE在GCC环境中的自我反思和任务推断过程。

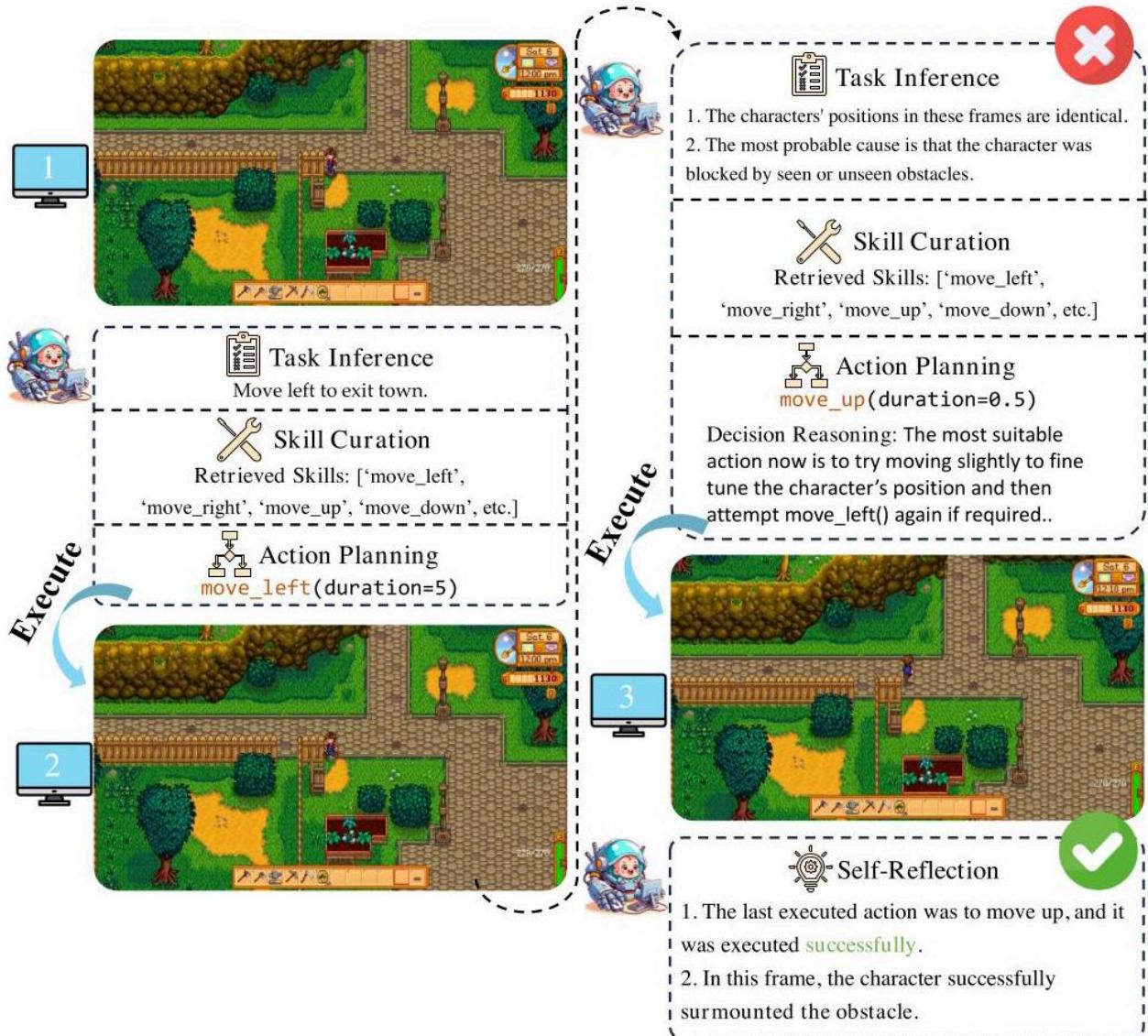


Figure 24: Case study of self-reflection on re-trying a failed task. Task instruction and context require the agent to exit town. A wrong direction is first selected, but the agent moves up after self-reflection. Only relevant modules are shown for better readability, though all modules (Figure 4) are executed per iteration.

图24：关于重试失败任务的自我反思案例研究。任务指令和背景要求智能体离开城镇。最初选择了错误方向，但智能体在自我反思后向上移动。为提高可读性，仅显示相关模块，尽管所有模块（图4）均在每次迭代中执行。

### 38.8.1 C.4.1 Self-Reflection

#### 38.8.2 C.4.1 自我反思

The Self-reflection module plays an important role in the completion of game missions in Stardew, giving our framework the ability to determine if the actions performed are complete and effective and to correct the errors of invalid actions. In the "Purchasing Seeds" task, the Agent is asked to return home from the store after purchasing items. At the "Home is on the left side of the store" prompt, the Agent controls the character to go left, but there are obstacles to keep going left, and the character must go up to circumnavigate the obstacles. As shown in the Figure 24, the role will initially be stuck at the obstacle and cannot continue to the left. Through Self-Reflection, the Agent can judge that it is currently in a state of obstruction, and moving to the left cannot be implemented smoothly. Therefore, the agent can adjust the direction upward to bypass the obstacle and enable the role to continue to the left until it returns home.

自我反思模块在《星露谷物语》(Stardew) 游戏任务完成中起着重要作用，使框架能够判断所执行动作是否完整有

效，并纠正无效动作的错误。在“购买种子”任务中，智能体被要求在购买物品后从商店返回家中。提示“家在商店左侧”时，智能体控制角色向左移动，但左侧有障碍物，角色必须向上绕行。如图24所示，角色最初被障碍物阻挡，无法继续向左。通过自我反思，智能体能判断当前处于阻塞状态，向左移动无法顺利进行，因此调整方向向上绕过障碍，继而继续向左直到回家。

### 38.8.3 C.4.2 Task-inference

#### 38.8.4 C.4.2 任务推断

Task Inference is a very effective module for completing game quests in Stardew. Its function is to decompose a vague and grand task into a specific sub-task, which effectively guides the Agent to complete the overall task. For example, in the Farming task, as shown in Figure 25, the task that the character needs to complete is "cultivate and harvest a parsnip." This is a complete but vague task. Through the Task Inference module, the Agent breaks down the task into (1) till the soil with the hoe, (2) plant the parsnip seeds, (3) water the planted seeds once daily for four days, (4) harvest the fully grown parsnip. This enables the Agent to know more clearly the steps needed to complete and finish the task successfully.

任务推断模块在完成《星露谷物语》游戏任务中非常有效。其功能是将模糊宏大的任务分解为具体子任务，有效指导智能体完成整体任务。例如，在农耕任务中，如图25所示，角色需要完成的任务是“种植并收获欧防风（parsnip）”。这是一个完整但模糊的任务。通过任务推断模块，智能体将任务细分为（1）用锄头耕地，（2）种植欧防风种子，（3）连续四天每天浇水一次，（4）收获成熟的欧防风。这样使智能体更清晰地了解完成任务所需的步骤，从而成功完成任务。

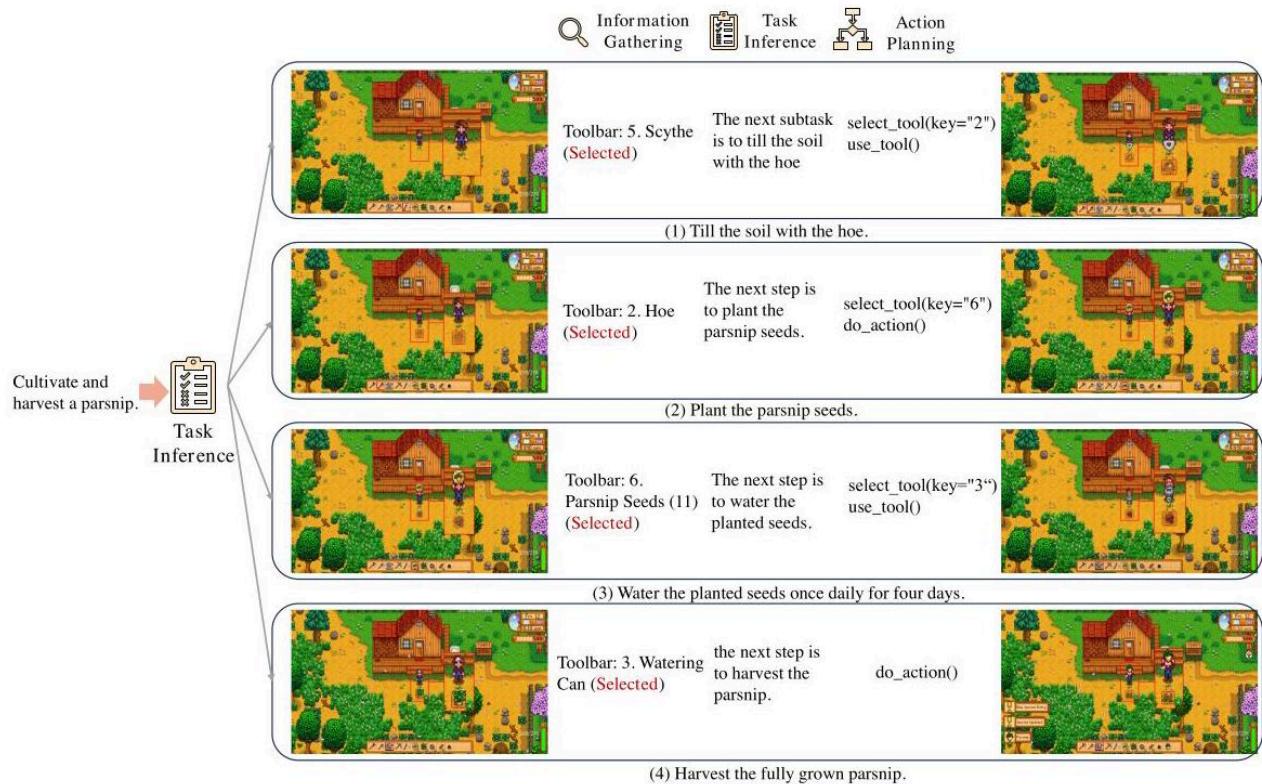


Figure 25: Case study of task inference on decomposing a task into specific sub-tasks. The complete task is to cultivate and harvest a parsnip. CRADLE decomposes the task into four sub-tasks by task inference. Only relevant modules are shown for better readability, though all modules (Figure 4) are executed per iteration.

图25：任务推理案例研究，将一个任务分解为具体的子任务。完整任务是种植和收获欧防风。CRADLE通过任务推理将任务分解为四个子任务。为提高可读性，仅显示相关模块，尽管所有模块（图4）均在每次迭代中执行。

## **38.9 C.5 Limitations of GPT-4o**

### **38.10 C.5 GPT-4o的局限性**

Fine-grained Control. Stardew Valley requires that players are positioned precisely to interact with objects, such as doors and NPCs. However, it is difficult for GPT-4o to take a pixel-level precise action. For example, GPT-4o can not take a precise movement even though the speed at which the figure moves is known. To alleviate this problem, we make some composite skills that use template-matching to complete some complex interaction tasks, such as purchasing items.

细粒度控制。《星露谷物语》(Stardew Valley)要求玩家精确定位以与物体(如门和NPC)互动。然而, GPT-4o难以执行像素级的精确操作。例如,尽管已知角色移动速度, GPT-4o仍无法实现精确移动。为缓解此问题,我们设计了一些复合技能,利用模板匹配完成诸如购买物品等复杂交互任务。

Perception in a 2D virtual world . In Stardew Valley, it's common for a character to be blocked by rocks or trees, and GPT-4o fails to tell if a character is blocked by looking at the image once, and can't predict if the next move will be blocked, which is very easy for a human to do by looking at the image. This indicates that GPT-4o is relatively weak in perceiving the virtual world in this game. In order to solve this problem, we compare the successive frames before and after in Self-Reflection to enable GPT-4o to judge the corresponding changes.

二维虚拟世界中的感知。在《星露谷物语》中,角色常被岩石或树木阻挡,GPT-4o仅凭一次图像无法判断角色是否被阻挡,也无法预测下一步移动是否会被阻挡,而人类通过观察图像很容易做到。这表明GPT-4o在感知该游戏虚拟世界方面相对较弱。为解决此问题,我们在自我反思阶段比较前后连续帧,使GPT-4o能够判断相应变化。

## **39 D Dealer's Life 2**

## **40 D Dealer's Life 2**

### **40.1 D.1 Introduction to Dealer's Life 2**

#### **40.2 D.1 Dealer's Life 2简介**

Dealer's Life 2 is a captivating indie simulation game developed by Abyte Entertainment. Renowned for its intricate negotiation mechanics and humorous portrayal of a pawn shop environment, the Dealer's Life 2是一款由Abyte Entertainment开发的引人入胜的独立模拟游戏。该游戏以其复杂的谈判机制和幽默的当铺环境描绘而闻名,

game is celebrated for its engaging gameplay that combines strategy with a quirky, cartoonish art style. As a simulation game with role-playing elements, Dealer's Life 2 is played from a first-person perspective, utilizing a mouse for point-and-click interactions and a keyboard for price inputs. This interface facilitates item appraisals, customer interactions, and comprehensive shop management.

游戏以策略与古怪卡通风格相结合的玩法著称。作为一款带有角色扮演元素的模拟游戏, Dealer's Life 2采用第一人称视角, 使用鼠标进行点选操作, 键盘输入价格。该界面便于物品鉴定、客户互动及全面的店铺管理。

In the game, players assume the role of a pawn shop manager, tasked with acquiring and selling various items to make a profit while managing their store's reputation and inventory. Players engage with a wide range of unique non-player characters (NPCs), each with their own distinct behaviors and negotiation styles. Whether bartering over the price of a rare collectible or managing unforeseen shop events, players must hone their haggling and strategic decision-making skills to succeed. Dealer's Life 2 operates in a closed-source format with no APIs available for accessing in-game data or automating gameplay functions. This setup ensures a hands-on experience where players are immersed in the day-to-day challenges of running a pawn shop. This game environment provides a unique and entertaining setting for testing the GCC's haggling and strategic decision-making abilities. We run our experiments using the latest version, V. 1.013\_W96 of the game.

在游戏中,玩家扮演当铺经理,负责收购和出售各种物品以盈利,同时管理店铺声誉和库存。玩家将与众多独特的非

玩家角色（NPC）互动，每个NPC拥有不同的行为和谈判风格。无论是为稀有收藏品讨价还价，还是应对突发店铺事件，玩家都需磨练讨价还价和战略决策能力以取得成功。Dealer's Life 2为闭源格式，未提供访问游戏数据或自动化游戏功能的API，确保玩家亲身体验经营当铺的日常挑战。该游戏环境为验证GCC的讨价还价及战略决策能力提供了独特且有趣的场景。我们使用游戏最新版本V.1.013\_W96进行实验。

### 40.3 D.2 Objectives

### 40.4 D.2 目标

We concentrate on evaluating the sustained management skills required to maximize profits through buying and selling a diverse range of items from customers. Therefore, the task in this game is defined as Weekly shop management, i.e., managing a shop for a week automatically. This game could effectively demonstrate the negotiation ability of the LMM in a trade and bargain. For example, giving an unacceptable price to the customers, i.e., a pretty low price for a seller customer or a very high price for a buyer customer, could cause the deal to fail directly, which brings no profit in this situation. The key is to carefully analyze the description of the item, e.g., the rarity and condition of the item, and more importantly, the response of the customer, i.e., the customer's mood changes.

我们重点评估通过买卖多样物品实现利润最大化所需的持续管理技能。因此，本游戏任务定义为每周店铺管理，即自动管理店铺一周。该游戏能有效展示大规模语言模型（LMM）在交易和讨价还价中的能力。例如，给客户不可接受的价格——对卖家客户报价过低或对买家客户报价过高——会直接导致交易失败，无法获利。关键在于仔细分析物品描述，如物品稀有度和状况，更重要的是客户的反应，即客户情绪变化。

Contrary to many games that feature detailed tutorials highlighting specific operations and objectives through each crucial step, Dealer's Life 2 does not provide such guidance. This absence transforms the game into a zero-shot, hard open-world task, where the LMM must directly apply its prior knowledge of haggling and strategic decision-making to a new and unfamiliar environment. To provide readers with a clear and straightforward understanding of the task, we illustrate the typical flow of a day's shop management through several key steps, presented in Table 10. 与许多游戏通过详细教程突出每个关键步骤的具体操作和目标不同，Dealer's Life 2不提供此类指导。这使游戏成为一个零样本、困难的开放世界任务，LMM必须直接将其讨价还价和战略决策的先验知识应用于全新且陌生的环境。为使读者清晰直观地理解任务，我们通过表10展示了一天店铺管理的典型流程及若干关键步骤。

Table 10: Key points in the open-ended mission, Weekly shop management in Dealer's Life 2. Figure 26 showcases snapshots of key points (specific sub-figures marked in parenthesis in the table).

表10：Dealer's Life 2中开放式任务“每周店铺管理”的关键点。图26展示了关键点的快照（表中括号标注具体子图）。

Task: Weekly shop management	Description
Open shop (Fig. 26a)	Start a new day shop management.
Dialog (Fig. 26b)	Choose an option in a dialog.
Item Description (Fig. 26c)	View the item information
Haggle (Fig. 26d)	Give a price for the item.
Deal Result (Fig. 26e)	View the deal results.
Stats (Fig. 26f)	View shop stats.

任务：每周商店管理	描述
开店（图26a）	开始新一天的商店管理。
对话（图26b）	在对话中选择一个选项。
物品描述（图26c）	查看物品信息
讨价还价（图26d）	为物品出价。
交易结果（图26e）	查看交易结果。
统计（图26f）	查看商店统计数据。

## 40.5 D.3 Implementation Details

### 40.6 D.3 实现细节

The implementation of Dealers' Life 2 also strictly follows the GCC framework, which includes Information Gathering, Self-Reflection, Task Inference, Skill Curation, Action Planning, and Action Execution. The details are described in Appendix A. Therefore, we emphasize the specific implementations for Dealers' Life 2.

《Dealer's Life 2》的实现同样严格遵循GCC框架，包括信息收集、自我反思、任务推断、技能策划、行动规划和行动执行。具体细节见附录A。因此，我们重点强调《Dealer's Life 2》的具体实现。

**Procedural Memory.** Due to the absence of a new-user guide, the LMM cannot directly and accurately know the operation method or effect of an action in the game, e.g., giving the price can only use the keyboard to input an integer in an abstract box in the bottom right of the haggle screen as shown in Figure 26d, by directly observing the screen. Unless the player executes an action and observes what is happening, the player cannot know what its effect is. However, this could easily cause severe errors in an open-world environment. For example, if the player gives a price at \$100,000 for an item without knowing what the box is, it could cause the player to lose all the money.

程序记忆。由于缺乏新手引导，语言大模型（LMM）无法直接且准确地了解游戏中某个动作的操作方法或效果，例如，给出价格只能通过键盘在讨价还价界面右下角的抽象框中输入整数，如图26d所示，需直接观察屏幕。除非玩家执行动作并观察发生了什么，否则无法知道其效果。然而，这在开放世界环境中极易导致严重错误。例如，若玩家在不知该框含义的情况下为某物品报价10万美元，可能导致玩家损失全部资金。

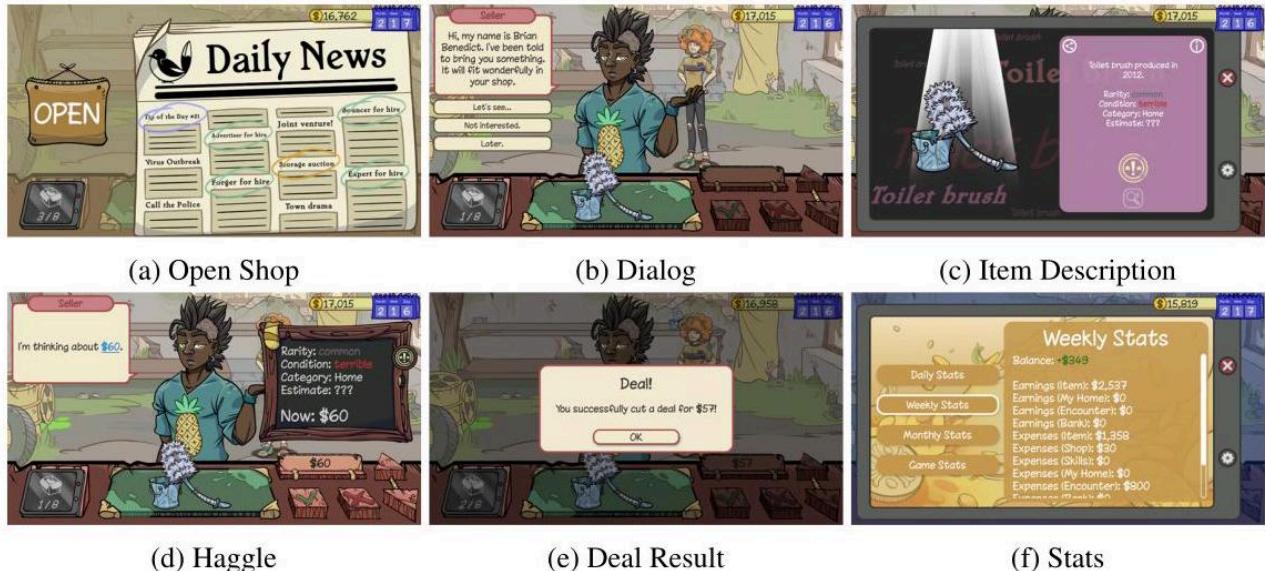


Figure 26: Image examples of key points in the open-ended task of Dealers' Life 2.

图26：《Dealer's Life 2》开放式任务关键点的图像示例。

Besides, this game is very simplified with finite types of screen content and fixed buttons positions for processing the deal, where we could categorize the screen types and design general atomic skills for them. Thus, with a focus on evaluating the LMM's zero-shot haggling and strategic decision-making ability in managing a shop, we believe it is reasonable to skip the skill curation by directly setting several atomic skills as the initialization of the procedural memory, such as "process\_dialog()" for clicking on the option of a dialog screen to keep the deal going on as shown in Figure 26b. The description of all the atomic skills is listed as follows:

此外，该游戏非常简化，屏幕内容类型有限，处理交易的按钮位置固定，我们可以对屏幕类型进行分类并设计通用的原子技能。因此，鉴于重点评估LMM在零样本讨价还价和店铺管理中的策略决策能力，我们认为直接将若干原子技

能设为程序记忆的初始化，跳过技能策划是合理的，例如“process\_dialog()”用于点击对话界面选项以继续交易，如图26b所示。所有原子技能描述如下：

- open\_shop(): The function to open the dealer's shop to start dealing for today.
- open\_shop(): 打开商店开始当天交易的功能。
- give\_price(price): The function to give a price for the item in the deal. The price must be an integer number.
- give\_price(price): 为交易物品报价的功能，价格必须为整数。
- process\_dialog(): The function to click on to choose the first option of the dialog to make the game go on.
- process\_dialog(): 点击选择对话框第一个选项以继续游戏的功能。
- close\_description\_page(): The function to close a description page showing information about the item details, daily stats, or the traits of the buyer or seller.
- close\_description\_page(): 关闭显示物品详情、每日统计或买卖双方特征信息的描述页面的功能。
- accept\_deal(): The function to click on the check mark to accept the deal on the confirmation dialog.
- accept\_deal(): 点击确认对话框中的勾选标记以接受交易的功能。
- reject\_deal(): The function to click on the cross mark to reject the deal on the confirmation dialog.
- reject\_deal(): 点击确认对话框中的叉号以拒绝交易的功能。
- finish\_buy(): The function to click on the ok button to finish the deal on the confirmation dialog.
- finish\_buy(): 点击确认对话框中的确定按钮以完成购买的功能。
- finish\_sell(): The function to click on the ok button to finish the selling on the confirmation dialog.
- finish\_sell(): 点击确认对话框中的确定按钮以完成销售的功能。

Self-Reflection. Additionally, as Dealers' Life 2 has no heavy need for a long-term reflection, so we only use the first and last frame of the video as input to reduce the number of tokens used per request. Finally, this self-reflection module could help to keep the game going, instead of sticking to the same point in the game.

自我反思。此外，由于《Dealer's Life 2》对长期反思需求不高，我们仅使用视频的首尾帧作为输入，以减少每次请求使用的token数量。最终，该自我反思模块有助于游戏继续进行，避免停滞在同一游戏点。

Action Planning. In this game, we restrict GPT-4 to output only one skill per action because it is a round-based game that does not require frequent execution of actions, and the state of the next time-step after an action is executed is highly uncertain, e.g., the unpredictable mood changes in a customer's response.

行动规划。在本游戏中，我们限制GPT-4每次行动仅输出一个技能，因为这是回合制游戏，不需要频繁执行动作，且动作执行后下一时间步的状态高度不确定，例如顾客反应中不可预测的情绪变化。

## 40.7 D.4 Case Studies

## 40.8 D.4 案例研究

Here we present a few game-specific case studies to further discuss CRADLE's reasoning and decision-making process in the GCC setting.

这里我们展示几个特定游戏的案例研究，以进一步讨论CRADLE在GCC环境中的推理和决策过程。



(a) Agent: Given that the customer is a seller and the item on offer is in bad condition with an estimated price of \$ 280, it would be wise to offer around \$150 to leave room for negotiation.



(b) Agent: Given the item's bad condition and \$280 estimate, A reasonable starting counteroffer would be 50-60% of its estimated value, and \$160 is a reasonable counteroffer.



(c) Agent: A reasonable next offer could be \$180 or \$190, aiming to reach a middle ground. I would still risk with \$160 to see if the offer is cut further.



(d) Agent: The customer is proposing her final offer at \$205. It will be necessary to accept the offer as this is still a profitable margin.

Figure 27: The reasoning of CRADLE in a successful deal with haggling. The price provided by CRADLE is keyed in by keyboard and mouse operations in the digital display box in the bottom-right corner.

图27展示了CRADLE与NPC卖家就一件价值280美元的物品进行成功谈判的过程。CRADLE通过考虑物品质量和客户的初始报价，确定了一个战略性的起始报价。在随后的谈判回合中，CRADLE利用其记忆保持报价接近最初评估的160美元，施压客户降低期望。然而，面对客户最终报价并表现出离开的倾向时，CRADLE也表现出灵活性，调整策略，从而达成一个仍然有利可图的最终协议。

#### 40.8.1 D.4.1 Successful Negotiation

#### 40.8.2 D.4.1 成功的谈判

Figure 27 illustrates a successful negotiation by CRADLE with an NPC seller over an item valued at \$280. CRADLE determines a strategic starting offer by considering both the item's quality and the customer's initial proposal. Throughout subsequent negotiation rounds, CRADLE leverages its memory to maintain an offer close to the initially assessed \$160, applying pressure on the customer to reduce their expectations. However, CRADLE also demonstrates flexibility, adapting its strategy when faced with the customer's final offer-signaled by their incline to leave. This allows CRADLE to secure a final agreement that still yields a profitable deal.

图27展示了CRADLE与NPC卖家就一件价值280美元的物品进行成功谈判的过程。CRADLE通过考虑物品质量和客户的初始报价，确定了一个战略性的起始报价。在随后的谈判回合中，CRADLE利用其记忆保持报价接近最初评估的160美元，施压客户降低期望。然而，面对客户最终报价并表现出离开的倾向时，CRADLE也表现出灵活性，调整策略，从而达成一个仍然有利可图的最终协议。

#### 40.8.3 D.4.2 Unsuccessful Negotiation

#### 40.8.4 D.4.2 失败的谈判

Figure 28 illustrates a scenario where CRADLE engages in an unsuccessful negotiation. The seller consistently demands a price above the estimated value of the item, while CRADLE, aiming to secure a profit, steadfastly offers a price below the estimated value. A common price cannot be arrived at after rounds of negotiation. Consequently, the negotiation fails to reach an agreement, resulting in the departure of the high-expectation customer.

图28展示了CRADLE参与的一次失败谈判场景。卖家始终坚持高于物品估价的价格，而CRADLE为了确保利润，坚定地提出低于估价的报价。经过多轮谈判，双方未能达成共识。结果，谈判失败，高期望的客户离开了。

#### 40.8.5 D.4.3 Acquiring and Selling of a Counterfeit Item

#### 40.8.6 D.4.3 购买和出售赝品

Figure 29 depicts a complete buying and selling cycle involving an item later identified as counterfeit by a buyer. Notably, CRADLE initially acquires the item without a known price estimate, basing its purchase solely on the seller's initial offer. In subsequent transactions, although CRADLE recognizes the item's potential forgery, it continues to trade the item in pursuit of maximizing profit. This strategy yields a substantial profit margin of 145%. However, it also poses a risk to the in-game character's reputation, highlighting a strategic decision made under the directive to prioritize profit.

图29描绘了一个完整的买卖循环，涉及一件后来被买家识别为赝品的物品。值得注意的是，CRADLE最初在没有已知价格估计的情况下，仅基于卖家的初始报价购买了该物品。在后续交易中，尽管CRADLE识别出该物品可能为赝品，但仍继续交易以追求最大利润。该策略带来了145%的可观利润率，但也对游戏角色的声誉构成风险，凸显了在优先考虑利润指令下做出的战略决策。



(a) Agent: I will offer an initial price of \$100 to reflect the poor condition while allowing room for negotiation since the condition is poor.



(b) Agent: The buyer suggests higher than the estimated price. I should offer a price lower than the item's estimated value. I will offer \$110.



(c) Agent: The seller has countered several times already, suggesting that further negotiations could result in a mutually acceptable price, possibly just a little bit below \$164.



(d) Agent: The negotiation failed. My next step is to acknowledge the failure and continue the game-play to interact with other customers to profit.

Figure 28: The reasoning of CRADLE in an unsuccessful deal with haggling. The price provided by CRADLE is keyed in by keyboard and mouse operations in the digital display box in the bottom-right corner.

图28：CRADLE在失败讨价还价交易中的推理过程。CRADLE提供的价格通过键盘和鼠标操作输入到右下角的数字显示框中。

## 40.9 D.5 Quantitative Evaluation

### 40.10 D.5 定量评估

The effectiveness of CRADLE in terms of game performance is evaluated through experiments conducted over a 7-day gaming scenario using nine quantitative financial metrics. A comprehensive description of these metrics is provided in Section D.6.

通过在为期7天的游戏场景中进行实验，使用九个定量财务指标评估CRADLE在游戏性能方面的有效性。关于这些指标的详细描述见D.6节。

### 40.11 D.6 Evaluation Metrics

#### 40.12 D.6 评估指标

Assuming the buying price for item  $i$  is denoted by  $B_i$ , the selling price by  $S_i$ , the market valuation by  $V_i$ , and the number of successfully traded items is  $n$ . To evaluate CRADLE's profitability and performance in Dealer's Life 2, we use the following evaluation metrics:

假设物品*i*的购买价格为 $B_i$ ，销售价格为 $S_i$ ，市场估价为 $V_i$ ，成功交易的物品数量为 $n$ 。为了评估CRADLE在《Dealer's Life 2》中的盈利能力和表现，我们使用以下评估指标：

- Turnover Rate (TR) can be calculated as the ratio of the number of successfully traded items to the total number of items considered (both successfully and unsuccessfully traded). It reflects the Agent's ability to successfully complete transactions and can indicate operational efficiency, market competitiveness, and negotiation effectiveness. The calculation formula is  $TR = \frac{n}{n+m}$ .
- 周转率 (TR) 可计算为成功交易物品数量与考虑的物品总数（包括成功和失败交易）的比率。它反映了代理完成交易的能力，并可指示运营效率、市场竞争力和谈判效果。计算公式为 $TR = \frac{n}{n+m}$ 。
- Gross Profit Margin (GPM) is the ratio of gross profit to sales revenue, reflecting the dealer's direct profit capability after selling items. The calculation formula is  $GPM = \frac{\sum_{i=1}^n S_i - B_i}{\sum_{i=1}^n S_i}$ .
- 毛利率 (GPM) 是毛利润与销售收入的比率，反映经销商售出物品后的直接盈利能力。计算公式为  

$$GPM = \frac{\sum_{i=1}^n S_i - B_i}{\sum_{i=1}^n S_i}$$
- Return on Investment (ROI) is the ratio of profit to investment, used to measure the dealer's return on investment for items. The calculation formula is  $ROI = \frac{\sum_{i=1}^n S_i - B_i}{\sum_{i=1}^n B_i}$ .
- 投资回报率 (ROI) 是利润与投资的比率，用于衡量经销商对物品的投资回报。计算公式为 $ROI = \frac{\sum_{i=1}^n S_i - B_i}{\sum_{i=1}^n B_i}$ 。
- Valuation Deviation (VD) reflects the difference between the selling price and the market valuation, used to evaluate the reasonableness of the pricing strategy. It is denoted as  $VD = \frac{\sum_{i=1}^n S_i - V_i}{\sum_{i=1}^n V_i}$ .
- 估值偏差 (Valuation Deviation, VD) 反映售价与市场估值之间的差异，用于评估定价策略的合理性。表示为

$$VD = \frac{\sum_{i=1}^n S_i - V_i}{\sum_{i=1}^n V_i}.$$

- Buying Price to Valuation Ratio (BPVR) can help determine whether the buying price is lower than the market valuation, reflecting the success of the procurement. The calculation formula is  $BPVR = \frac{\sum_{i=1}^n B_i}{\sum_{i=1}^n V_i}$ .
- 采购价与估值比率（Buying Price to Valuation Ratio, BPVR）有助于判断采购价是否低于市场估值，反映采购的成功与否。计算公式为 $BPVR = \frac{\sum_{i=1}^n B_i}{\sum_{i=1}^n V_i}$ .
- Selling Price to Valuation Ratio (SPVR) reflects the selling price relative to the market valuation, helping to assess the success of the sales. The calculation formula is  $SPVR = \frac{\sum_{i=1}^n S_i}{\sum_{i=1}^n V_i}$ .
- 售价与估值比率（Selling Price to Valuation Ratio, SPVR）反映售价相对于市场估值的情况，有助于评估销售的成功。计算公式为 $SPVR = \frac{\sum_{i=1}^n S_i}{\sum_{i=1}^n V_i}$ .



(a) Agent: Given that we do not know the estimated price, I proceed with 70%-80% of the asked price, around \$4,200 to \$4,480.

(a) 代理：由于我们不知道估价，我按要价的70%-80%进行，大约在4200美元到4480美元之间。

(b) Agent: Since the seller is willing to lower the price, the current counteroffer may still be negotiated.

(b) 代理：既然卖家愿意降价，目前的还价仍有谈判空间。



(c) Agent: The counteroffer of \$4,900 seems reasonable, but I can go further for the best possible price.

(c) 代理：4900美元的还价看起来合理，但我可以继续争取更好的价格。

(d) Agent: The response will be final. Given this, accepting the deal is a sensible choice to secure the item.

(d) 代理：这是最终回复。鉴于此，接受交易是确保获得该物品的明智选择。



(e) Agent: As I am selling the item, I'll try for a higher price for higher profits.

(e) 代理：既然我是卖方，我会争取更高的价格以获得更大利润。

(f) Agent: I note that the customer rejected my counter-offer, calling the item fake.

(f) 代理：我注意到客户拒绝了我的还价，称该物品为赝品。





(g) Agent: The seller is in an urgent need to buy the jeweled necklace.

(g) 代理：卖家急需购买这条镶宝石的项链。

(h) Agent: From the warning window, I am told that this necklace is fake.

(h) 代理：从警告窗口得知这条项链是赝品。





(i) Agent: Though the jeweled necklace is fake, accepting this offer provides substantial profit with the buyer's offer at \$11,400.

(i) 代理：虽然这条镶宝石的项链是赝品，但接受买方11400美元的报价仍能带来可观利润。

(j) Agent: Despite the successful sale at a good price, the item is revealed as a fake. The added profit is good (+145%)

(j) 代理：尽管以好价成功售出，但该物品被证实为赝品。额外利润可观 (+145%)。

Figure 29: Case in acquiring and selling an item for multiple attempts with reasoning, and dealing with unexpected information on the authenticity. The price provided by CRADLE is keyed in by keyboard and mouse operations in the digital display box in the bottom-right corner.

图29：多次尝试获取和出售物品的案例，包含推理过程及应对真实性意外信息。CRADLE提供的价格通过数字显示框右下角的键盘和鼠标操作输入。

- Average Profit Rate (APR) reflects the overall profitability of the dealer on items. Assuming the return rate for item  $i$  is  $\frac{S_i - B_i}{B_i}$ , the calculation formula of average return rate is denoted as  $APR = \frac{1}{n} \sum_{i=1}^n \frac{S_i - B_i}{B_i}$ .
- 平均利润率 (Average Profit Rate, APR) 反映经销商对物品的整体盈利能力。假设物品*i*的回报率为 $\frac{S_i - B_i}{B_i}$ ，平均回报率的计算公式表示为 $APR = \frac{1}{n} \sum_{i=1}^n \frac{S_i - B_i}{B_i}$ 。
- Maximum Return Rate (MRR) is the highest return rate among all items. The calculation formula is  $MRR = \max \left( \frac{S_1 - B_1}{B_1}, \frac{S_2 - B_2}{B_2}, \dots, \frac{S_n - B_n}{B_n} \right)$ .
- 最大回报率 (Maximum Return Rate, MRR) 是所有物品中最高的回报率。计算公式为  $MRR = \max \left( \frac{S_1 - B_1}{B_1}, \frac{S_2 - B_2}{B_2}, \dots, \frac{S_n - B_n}{B_n} \right)$ 。
- Minimum Return Rate (mRR) is the lowest return rate among all items. The calculation formula is  $mRR = \min \left( \frac{S_1 - B_1}{B_1}, \frac{S_2 - B_2}{B_2}, \dots, \frac{S_n - B_n}{B_n} \right)$ .
- 最低回报率 (mRR) 是所有物品中最低的回报率。计算公式为 $mRR = \min \left( \frac{S_1 - B_1}{B_1}, \frac{S_2 - B_2}{B_2}, \dots, \frac{S_n - B_n}{B_n} \right)$ 。

Table 11: Performance of CRADLE with GPT-4o in Dealer's Life 2 gameplay. "# attempts" represents the total number of all negotiation attempts on items, including both successful and unsuccessful transactions.

表11：CRADLE结合GPT-4o在《Dealer's Life 2》游戏中的表现。“#尝试次数”表示所有物品的谈判总次数，包括成功和失败的交易。

Exp	#attempts	TR↑	GPM↑	ROI↑	VD↑	BPVR	\$\downarrow\$	SPVR↑	APR↑	MRR↑	mRR↑
01	13	92.86	20.38	25.60	13.17	90.10		113.17	42.97	105.56	0.00
02	12	91.67	18.89	23.30	23.30	100.00		123.30	17.98	97.76	0.00
03	12	83.33	26.81	36.63	34.39	98.36		134.39	38.68	127.27	-8.06
04	9	100.00	49.35	87.45	80.69	93.53		165.74	66.45	145.16	0.00
05	12	100.00	20.61	25.25	25.25	100.00		125.25	23.08	44.33	0.00
Avg.	11.6	93.57	27.21	39.65	35.36	96.40		132.37	37.83	104.02	-1.61
<hr/>											
实验	尝试次数	TR↑	GPM↑	ROI↑	VD↑	BPVR	\$\downarrow\$	SPVR↑	APR↑	MRR↑	mRR↑
01	13	92.86	20.38	25.60	13.17	90.10		113.17	42.97	105.56	0.00
02	12	91.67	18.89	23.30	23.30	100.00		123.30	17.98	97.76	0.00
03	12	83.33	26.81	36.63	34.39	98.36		134.39	38.68	127.27	-8.06
04	9	100.00	49.35	87.45	80.69	93.53		165.74	66.45	145.16	0.00
05	12	100.00	20.61	25.25	25.25	100.00		125.25	23.08	44.33	0.00
平均	11.6	93.57	27.21	39.65	35.36	96.40		132.37	37.83	104.02	-1.61

## 41 E Cities: Skylines

### 42 E 城市：天际线

#### 42.1 E.1 Introduction to Cities: Skylines

#### 42.2 E.1 城市：天际线简介

Cities: Skylines is a single-player open-ended city-building simulation game developed by Colossal Order. In the game, players assume the role of a city planner, tasked with building and managing various aspects of a city to ensure its growth and prosperity. Players engage with a wide range of urban challenges, from managing traffic flow to balancing the budget, and from providing essential services to fostering a vibrant economy. Each decision impacts the city's development, requiring players to hone their planning and strategic decision-making skills to succeed. Effective city management leads to thriving neighborhoods, a growing economy, and high citizen satisfaction, while mismanagement can result in traffic congestion, service shortages, and a decline in population and reputation. Proper planning and responsive governance are crucial for a city that flourishes and remains appealing to its residents and visitors.

《城市：天际线》是一款由Colossal Order开发的单人开放式城市建设模拟游戏。在游戏中，玩家扮演城市规划者的角色，负责建设和管理城市的各个方面，以确保其增长和繁荣。玩家需要应对各种城市挑战，从管理交通流量到平衡预算，从提供基本服务到促进经济活力。每一个决策都会影响城市的发展，要求玩家磨练规划和战略决策能力以取得成功。有效的城市管理能够带来繁荣的社区、不断增长的经济和高市民满意度，而管理不善则可能导致交通拥堵、服务短缺、人口和声誉下降。合理规划和灵活治理是城市繁荣并保持对居民和游客吸引力的关键。

As the city's infrastructure and various supporting resources are well-developed, it can attract more people. And a larger population brings more tax revenue and also brings greater expenses to the city's operations. If operated properly, the increasing population can continuously unlock richer urban facilities; if operated improperly, such as road congestion, insufficient services, housing shortage, water and electricity shortage, noise pollution, water pollution, excessive garbage, disease, fire Situation, etc., will all lead to population decline.

随着城市基础设施和各种配套资源的完善，城市能够吸引更多人口。更大的人口带来更多税收收入，同时也增加了城市运营的开支。如果运营得当，人口增长可以持续解锁更丰富的城市设施；若运营不当，如道路拥堵、服务不足、住房短缺、水电供应不足、噪音污染、水污染、垃圾过多、疾病、火灾等情况，都会导致人口减少。

This game could be used to evaluate agents' strategies in managing urban development and resource allocation. By simulating different scenarios, agents can experiment with various policies and infrastructural changes to see their impacts on the city's growth and sustainability. Effective strategies may involve optimizing public transportation systems to reduce road congestion, investing in renewable energy sources to prevent power shortages, and implementing comprehensive waste management programs to handle excessive garbage. It offers a risk-free environment to test innovative ideas and learn from the consequences of their actions, ultimately promoting a deeper understanding of sustainable urban development.

该游戏可用于评估智能体在城市发展和资源分配管理中的策略。通过模拟不同场景，智能体可以尝试各种政策和基础设施变更，观察其对城市增长和可持续性的影响。有效策略可能包括优化公共交通系统以减少道路拥堵，投资可再生能源以防止电力短缺，以及实施全面的垃圾管理计划以处理过量垃圾。它提供了一个无风险的环境来测试创新想法并从行动后果中学习，最终促进对可持续城市发展的深入理解。

Though this game is ranked very positive on Steam, it is notorious for its extremely high difficulty for beginners, as it lacks a detailed tutorial in the beginning, which introduces more challenges for CRADLE to deal with. On the other side, Although the successor, Cities: Skylines 2, simplified the controls and provided a detailed tutorial for beginners, it became notorious for poor optimization and frequent crashes that caused computer blue screens. As a result, we had to back to using Cities: Skylines 1 instead of 2. And we do not apply any modes to the game. We use the latest version of the game (version 1.17.1-f4).

尽管该游戏在Steam上评价非常正面，但因缺乏详细的新手教程而以极高的入门难度著称，这给CRADLE带来了更多挑战。另一方面，虽然续作《城市：天际线2》简化了操作并为新手提供了详细教程，但因优化差和频繁崩溃导致蓝屏而声名狼藉。因此，我们不得不回归使用《城市：天际线1》而非2，并且未对游戏应用任何模组。我们使用的是游戏的最新版本（版本1.17.1-f4）。

### 42.3 E.2 Objectives

### 42.4 E.2 目标

Our mission is to build cities so that they can support as many people as possible. Maps in this game are usually very large, which usually costs human players dozens of hours to cover all areas. Besides, the technology tree unlocks as the population grows, which requires multiple turns of planning and building. In this work, we simplified the problem by starting the game near the water and fixing the viewpoint (as shown in Figure 30), so that CRADLE can leverage the pixel position in the screenshot to locate the position of placed buildings and facilities. Agents start with a plot of land, which is equipped with an entry and an exit from a major highway, providing crucial access for future traffic flow, and proximity to the water source, which is essential for the city's water supply needs. And we focus on the first turn of planning, i.e., pause the game and stop the passage of the in-game time, use the initial starting funds of C70,000 and the most basic road, water, and electricity facilities provided at the beginning of the game, which is enough to achieve the first milestone, Little Hamlet with the population of 440 in the game. Then what kind of city can CRADLE create? Can this city ensure water and electricity supply to keep functioning normally while reasonably dividing residential, commercial, and industrial zones? A run is terminated when it reaches the maximal steps, 1000, or the budget is used up (less than C 1000).

我们的任务是建设能够容纳尽可能多人口的城市。游戏中的地图通常非常大，普通玩家通常需要数十小时才能覆盖所有区域。此外，科技树会随着人口增长而解锁，需要多轮规划和建设。在本工作中，我们简化问题，从靠近水源的位置开始游戏并固定视角（如图30所示），以便CRADLE利用截图中的像素位置定位已建建筑和设施的位置。智能体从一块地皮开始，该地块配备了通往主要高速公路的入口和出口，为未来的交通流提供关键通道，并靠近水源，满足城市的供水需求。我们关注第一轮规划，即暂停游戏并停止游戏内时间流逝，使用初始资金7万C和游戏开始时提供的最基本的基础设施，这足以实现游戏中的第一个里程碑“小村庄”，人口为440。CRADLE能建造出怎样的城市？该城市能否确保水电供应正常运行，同时合理划分住宅、商业和工业区？当达到最大步数1000或预算用尽（低于1000C）时，运行终止。



Figure 30: Demonstration for the initialization location of our mission in City: Skylines, which is near the river and contains the entry and exit of the highways.

图30：我们在《城市：天际线》中任务初始化位置的示意图，位于河流附近，包含高速公路的入口和出口。

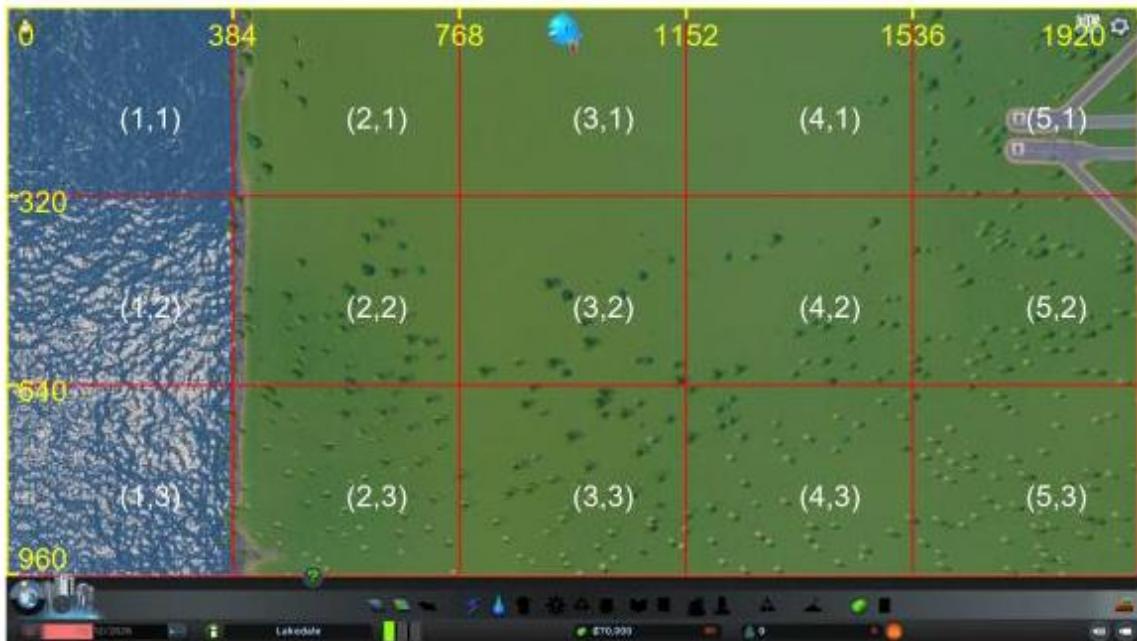


Figure 31: Visual prompting methods we use for Cities: Skylines. The full screenshot is divided into  $3 \times 5$  grids and each grid is assigned a unique white coordinate.

图31：我们在《城市：天际线》中使用的视觉提示方法。完整截图被划分为 $3 \times 5$ 个网格，每个网格分配一个唯一的白色坐标。

## 42.5 E.3 Evaluation Metric

### 42.6 E.3 评估指标

To measure the completeness of the city built by the agent, we design the following preliminary metrics:

为了衡量智能体建造城市的完整性，我们设计了以下初步指标：

- Roads in closed loop: Whether the road is a closed loop, which is crucial for ensuring smooth traffic flow and is beneficial for the city's future development.  
• 闭合环路的道路：道路是否形成闭合环路，这对于确保交通流畅至关重要，有利于城市的未来发展。
- Sufficient water supply: To ensure a sufficient water supply, the player needs to construct a water pumping station at the shoreline and then use water pipes to cover every district along the roads. To manage the effluent effectively, the other end of the water pipe network must be equipped with the water drain pipe which is also required to be placed near the shoreline.  
• 充足的供水：为了确保充足的供水，玩家需要在海岸线建造水泵站，然后使用水管沿道路覆盖每个区域。为了有效管理排放，水管网络的另一端必须配备排水管，且排水管也需放置在靠近海岸线的位置。
- Sufficient electricity supply: Both zones and water facilities need electricity to power. To provide sufficient electricity supply, the player can build a coal power plant or wind turbine. Considering coal power plants cost too much and will create heavy pollution, wind turbines combined with the power lines are a better choice at the beginning. The electricity area extends automatically based on the presence of buildings and infrastructure that consume electricity.  
• 充足的电力供应：区域和水务设施都需要电力支持。为了提供充足的电力供应，玩家可以建造燃煤电厂或风力发电机。考虑到燃煤电厂成本高且会造成严重污染，风力发电机结合电力线路是初期更好的选择。电力覆盖范围会根据建筑和基础设施的存在自动扩展，这些建筑和设施会消耗电力。
- Zones Area > 90%: The built two-lane road will provide empty space for the development of zones, i.e., residential zone, commercial zone and industrial zone. Residential zones provide houses for people to live in, which is the most essential zone to increase the population. Commercial zones provide places for small businesses, shops, and services produced in the industrial zones or imported. Industrial zones provide jobs for the residents and products for commercial buildings, which is also important to attract more people to move to the city. This metric is used to evaluate whether 90% of the available areas are covered by the zones. The agent needs to reasonably allocate the areas and proportions of various zones to achieve better city development and attract a larger population.  
• 区域面积 > 90%：已建成的双车道道路将为区域发展提供空地，即住宅区、商业区和工业区。住宅区为居民提供居住场所，是增加人口的最关键区域。商业区为小型企业、商店及工业区生产或进口的服务提供场所。工业区为居民提供就业岗位和为商业建筑提供产品，这对于吸引更多人口迁入城市也非常重要。该指标用于评估可用区域中有90%被区域覆盖。代理需要合理分配各类区域的面积和比例，以实现更好的城市发展并吸引更多人口。
- Maximal population: After CRADLE finishes building, we will unpause the game and start the simulation. Then houses start to be built and residents start to move in. We will record the maximal population during the simulation as the value for this metric.
- 最大人口：CRADLE完成建造后，我们将取消暂停游戏并开始模拟。随后开始建造住宅，居民开始迁入。我们将在模拟过程中记录最大人口作为该指标的数值。
- Maximal population with human assistance: We find that cities built by CRADLE manage to meet most of the requirements but suffer a significant population loss due to a few easy-to-fix mistakes. So after CRADLE finishes the design of the city, we apply human assistance that attempts to address these small mistakes within 3 unit operations (building or removing a road/facility/a place of zones is counted as one unit operation). We will also record the maximum population during the simulation in the city with human assistance.
- 人工辅助下的最大人口：我们发现CRADLE建造的城市能够满足大部分需求，但由于一些容易修正的小错误导

致人口显著流失。因此，在CRADLE完成城市设计后，我们进行人工辅助，尝试在3个单位操作内（建造或拆除道路/设施/区域位置均计为一个单位操作）修正这些小错误。我们也将 在人工辅助的城市模拟过程中记录最大人口。

## 42.7 E.4 Implementation Details

### 42.8 E.4 实现细节

The implementation of Cities: Skylines also strictly follows the GCC framework, which includes Information Gathering, Self-Reflection, Task Inference, Skill Curation, Action Planning and Action Execution. The details are described in Appendix A. Therefore, we emphasize the specific design for Cities: Skylines.

Cities: Skylines的实现同样严格遵循GCC框架，包括信息收集、自我反思、任务推断、技能策划、行动规划和行动执行。具体细节见附录A。因此，我们重点强调Cities: Skylines的具体设计。

Pause. Since the game is stopped before starting the simulation, there is no need to unpause and pause the game while executing actions.

暂停。由于游戏在开始模拟前处于暂停状态，执行操作时无需取消暂停或再次暂停游戏。

Visual Prompting. As shown in Figure 31, similar to Stardew Valley, we divide each screenshot into  $3 \times 5$  grids with an axis based on the resolution of the game screen. Then CRADLE can utilize the pixel-level position in the screenshot to locate the building and facility. We empirically find that this visual prompting method can result in a more precise control of GPT-4o.

视觉提示。如图31所示，类似于Stardew Valley，我们将每张截图根据游戏屏幕分辨率划分为 $3 \times 5$ 个网格。

CRADLE可以利用截图中的像素级位置来定位建筑和设施。经验表明，这种视觉提示方法能使GPT-4o实现更精确的控制。

Information Gathering. In Cities: Skylines, the game's perspective is typically adjustable, allowing players to zoom in and out, rotate, and pan across their cityscape to get a detailed view of their urban development. To ensure consistency and ease of navigation for GPT-4o, we have locked the camera angle and applied a visual prompting method to enhance GPT-4o's visual understanding. Besides, we use GPT-4o to extract key information, such as budget, population, construction information and error messages, in the game.

信息收集。在Cities: Skylines中，游戏视角通常可调，允许玩家缩放、旋转和平移城市景观以获得详细视图。为确保GPT-4o的一致性和便捷导航，我们锁定了摄像机角度，并采用视觉提示方法增强GPT-4o的视觉理解。此外，我们使用GPT-4o提取游戏中的关键信息，如预算、人口、建设信息和错误信息。

It is worth noting that in this module, we feed the original screenshot to GPT-4o, rather than the augmented screenshot with axis and coordinates. We find that the numbers and lines may cover some key information and result in wrong OCR recognition. For example, the construction information, "Estimated Production: 120,000m<sup>3</sup>/week" may be mistakenly interpreted as "Estimated Production: 000,000m<sup>3</sup>/week" by GPT-4o, due to interference from the lines and numbers. This construction information is a key signal for the suitable place of the water pumping station. For the other modules, we feed GPT-4o with the augmented screenshots.

值得注意的是，在此模块中，我们向GPT-4o输入的是原始截图，而非带有坐标轴和坐标的增强截图。我们发现数字和线条可能遮挡关键信息，导致OCR识别错误。例如，建设信息“预计产量：120,000立方米/周”可能被GPT-4o误识为“预计产量：000,000立方米/周”，这是由于线条和数字的干扰。该建设信息是确定水泵站合适位置的关键信号。其他模块则输入增强截图给GPT-4o。

Self-Reflection. Since actions in this game are very short, and each of them has a significant effect shown in the last screenshot. We only use the first screenshot and the last screenshot of the video clip as input to this module, which is proved to be enough for not missing any important information.

自我反思。由于游戏中的操作非常简短，且每个操作的效果都在最后一张截图中明显体现，我们仅使用视频片段的第一张和最后一张截图作为该模块的输入，事实证明这足以捕捉所有重要信息。

Task Inference. Due to the lack of a detailed tutorial, we have to provide a draft blueprint for the GPT-40 as the plan at the beginning to help GPT-40 to determine the next step to do. This plan provides guidance to the orders of building each facility and how to build a closed road, how to ensure water and electricity supply and zone placement. Even so, we find that GPT-40 failed frequently to follow the plan, resulting in the lack of building some important facilities, like water pumping stations.

任务推断。由于缺乏详细的教程，我们不得不在一开始为GPT-40提供一个草案蓝图作为计划，帮助GPT-40确定下一步的行动。该计划指导了各设施的建设顺序、如何建造封闭道路、如何确保水电供应及区域划分。即便如此，我们发现GPT-40经常未能遵循计划，导致缺少一些重要设施的建设，如水泵站。

Skill Curation. Due to the lack of detailed tutorials in the game, we generate the skills through self-exploration in this game. The skill generation basically involves manipulating the toolbar to understand the items on it. The pseudo-code for skill generation is described in Algorithm 1. This process leverages SAM for objective grounding and GPT-40 to gather information about the objects provided by the game, subsequently generating skills based on a predefined template. An example of the process is shown in Fig 32, 33, 34, 35, 36 and 37.

技能策划。由于游戏中缺乏详细教程，我们通过自我探索生成技能。技能生成基本涉及操作工具栏以理解其上的项目。技能生成的伪代码描述见算法1。该过程利用SAM进行目标定位，GPT-40收集游戏提供的对象信息，随后基于预定义模板生成技能。该过程示例见图32、33、34、35、36和37。



Figure 32: The toolbar in Cities: Skylines

图32: Cities: Skylines中的工具栏

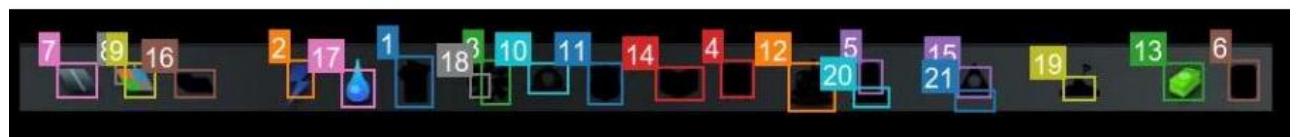


Figure 33: The grounding result of the toolbar in Cities: Skylines

图33: Cities: Skylines中工具栏的定位结果



Figure 34: When hovering the mouse over a toolbar item, the pop-up description is "Water & Sewage". The skill generated is then called "open\_water\_sewage\_menu".

图34：当鼠标悬停在工具栏项目上时，弹出描述为“水与污水”。生成的技能称为“open\_water\_sewage\_menu”。



Figure 35: When hovering the mouse over a toolbar item, the pop-up description is "Education - Reach a population of 440". As this is not selectable for now, GPT-4o does not generate a new skill for it.

图35：当鼠标悬停在工具栏项目上时，弹出描述为“教育 - 达到440人口”。由于当前不可选择，GPT-40未为其生成新技能。

Action Planning. In this game, we only let GPT-4o output one skill for each action since we observe that GPT-40 tends to output try\_place and confirm placement together if we allow it to output and execute multiple skills in one action, which is against the intention of our design for the try\_place action.

行动规划。在本游戏中，我们仅允许GPT-40每次行动输出一个技能，因为观察到若允许其在一次行动中输出并执行多个技能，GPT-40倾向于同时输出try\_place和确认放置，这违背了我们对try\_place动作设计的初衷。

Procedure Memory. Skills generated through self-exploration are listed below:

过程记忆。通过自我探索生成的技能如下：

- open\_roads\_menu(): The function to open the roads options in the lower menu bar for further determination of which types of roads to build.  
• open\_roads\_menu(): 打开下方菜单栏中的道路选项，以进一步确定建设哪种类型的道路。
- open\_electricity\_menu(): The function to open the electricity options in the lower menu bar for further determination of which types of power facility to build.  
• open\_electricity\_menu(): 打开下方菜单栏中的电力选项，以进一步确定建设哪种类型的电力设施。
- open\_water\_sewage\_menu(): The function to open the water and sewage options in the lower menu bar for further determination of which types of water and sewage to build.  
• open\_water\_sewage\_menu(): 打开下方菜单栏中的水与污水选项，以进一步确定建设哪种类型的水与污水设施。

- `open_zoning_menu()`: The function to open the zoning options in the lower menu bar for further determination of which types of zonings to build.
- `open_zoning_menu()`: 打开下方菜单栏中的分区选项，以进一步确定建设哪种类型的分区。
- `try_place_two_lane_road ( $x_1, y_1, x_2, y_2$ )` : Previews the placement of a road between two specified points,  $(x_1, y_1)$  and  $(x_2, y_2)$  ,with  $x_1, y_1$  being the coordinate of start point of the road, and  $(x_2, y_2)$  being the coordinate of end point of the road. This function does not actually construct the road, but rather displays a visual representation of where the road would be placed if confirmed.
- `try_place_two_lane_road ( $x_1, y_1, x_2, y_2$ )` : 预览在两个指定点 $(x_1, y_1)$ 和 $(x_2, y_2)$ 之间放置一条双车道道路， $x_1, y_1$ 为道路起点坐标， $(x_2, y_2)$ 为终点坐标。该函数不实际建造道路，仅显示若确认放置，道路将被放置的位置的视觉效果。
- `try_place_wind_turbine ( $x, y$ )` : Previews the placement of a wind turbine on point,  $(x, y)$  . This function does not actually construct the wind turbine, but rather displays a visual representation of where the wind turbine would be placed if confirmed.
- `try_place_wind_turbine ( $x, y$ )` : 预览在点 $(x, y)$ 放置风力涡轮机。该函数不实际建造风力涡轮机，仅显示若确认放置，风力涡轮机将被放置的位置的视觉效果。
- `try_place_water_pumping_station ( $x, y$ )` : Previews the placement of a water pumping station on point,  $(x, y)$  . This function does not actually construct the water pumping station, but rather displays a visual representation of where the water pumping station would be placed if confirmed.
- `try_place_water_pumping_station ( $x, y$ )` : 预览在点 $(x, y)$ 放置水泵站。该函数不实际建造水泵站，仅显示若确认放置，水泵站将被放置的位置的视觉效果。
- `try_place_water_pipe ( $x_1, y_1, x_2, y_2$ )` : Previews the placement of a water pipe between two specified points,  $(x_1, y_1)$  and  $(x_2, y_2)$  . This function does not actually construct the water pipe, but rather displays a visual representation of where the water pipe would be placed if confirmed.
- `try_place_water_pipe ( $x_1, y_1, x_2, y_2$ )` : 预览在两个指定点 $(x_1, y_1)$ 和 $(x_2, y_2)$ 之间放置水管。该函数不实际建造水管，仅显示若确认放置，水管将被放置的位置的视觉效果。
- `try_place_water_drain_pipe ( $x, y$ )` : Previews the placement of a water drain pipe on point,  $(x, y)$  . This function does not actually construct the water drain pipe, but rather displays a visual representation of where the water drain pipe would be placed if confirmed.
- `try_place_water_drain_pipe ( $x, y$ )` : 预览在点 $(x, y)$ 上放置排水管的位置。此功能并不实际构建排水管，而是显示如果确认放置，排水管将被放置的位置的可视化表示。

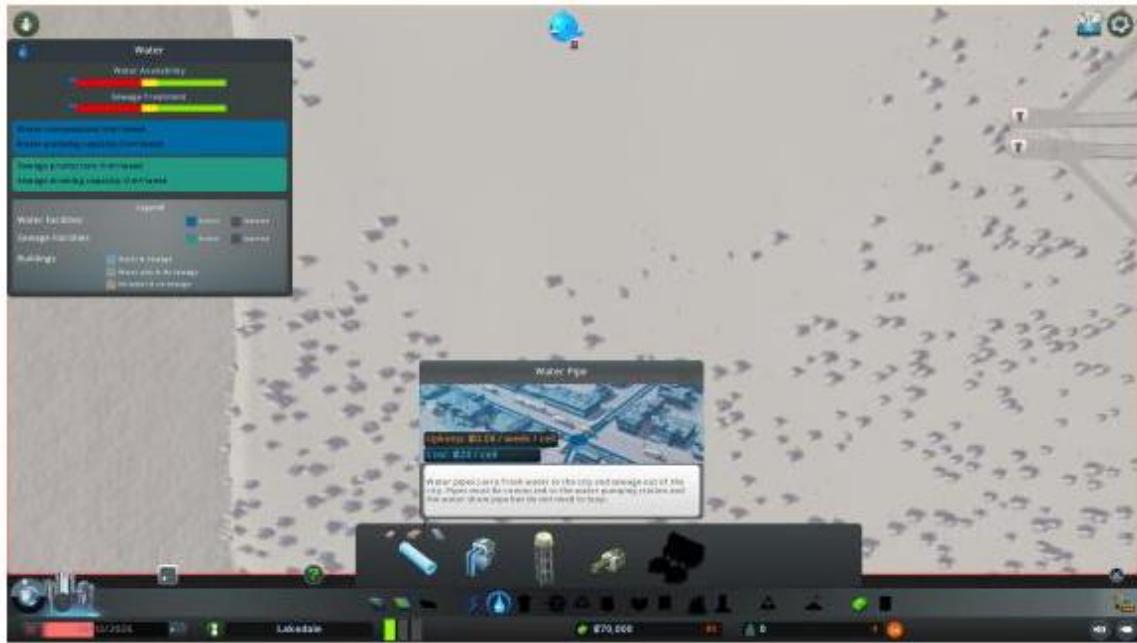


Figure 36: The Water & Sewage menu is opened by executing the new skill "open\_water\_sewage\_menu". The Agent then hovers the mouse over a second-level tool-bar item, the pop-up description is "Water Pipe", and the generated skill is called "try\_place\_water\_pipe".

图36：通过执行新技能“open\_water\_sewage\_menu”打开水与污水菜单。代理随后将鼠标悬停在二级工具栏项上，弹出描述为“水管”，生成的技能称为“try\_place\_water\_pipe”。

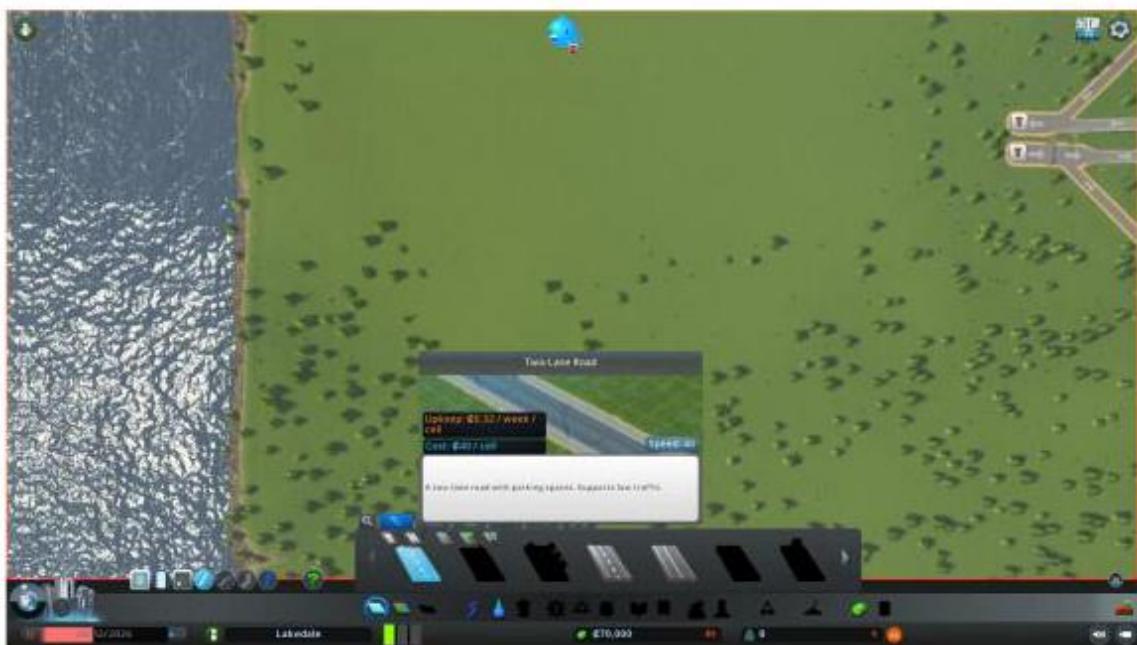


Figure 37: The Roads menu is opened by executing the new skill "open\_roads\_menu". The Agent then hovers the mouse over a second-level toolbar item, the pop-up description is "Two-Lane Road", and the generated skill is called "try\_place\_two\_lane\_road".

图37：通过执行新技能“open\_roads\_menu”打开道路菜单。代理随后将鼠标悬停在二级工具栏项上，弹出描述为“双车道道路”，生成的技能称为“try\_place\_two\_lane\_road”。

Algorithm 1: Skill Generation

算法1：技能生成

---

Input: Toolbar with objects, Skill template

输入：带有对象的工具栏，技能模板

Output: Procedure memory with generated skills

输出：包含生成技能的过程内存

Initialize procedure memory;

初始化过程内存；

for each object in the toolbar do

对工具栏中的每个对象执行

```
1 Hover the mouse on the object to get the description;  
2 将鼠标悬停在对象上以获取描述;  
  
1 Generate skill using GPT-4o based on the object description and the skill template;  
2 基于对象描述和技能模板，使用GPT-4o生成技能;  
  
1 Store generated skill in procedure memory;  
2 将生成的技能存储在过程内存中;  
  
1 Execute the generated skill to enter the second-level toolbar;  
2 执行生成的技能以进入二级工具栏;  
  
1 for each object in the second-level toolbar do  
2 对二级工具栏中的每个对象执行  
  
1 Hover the mouse on the object to get the description;  
2 将鼠标悬停在对象上以获取描述;  
  
1 Generate skill using GPT-4o based on the object description and skill template;  
2 基于对象描述和技能模板，使用GPT-4o生成技能;  
  
1 Store generated skill in procedure memory;  
2 将生成的技能存储在过程内存中;
```

return procedure memory

返回过程内存

---

- `try_place_commercial_zone ( $x_1, y_1, x_2, y_2$ )` : Previews the placement of a commercial zone within a rectangular region with diagonal corners at  $(x_1, y_1)$  and  $(x_2, y_2)$ . This function does not actually construct the commercial zone, but rather displays a visual representation of where the commercial zone would be placed if confirmed.
- `try_place_commercial_zone ( $x_1, y_1, x_2, y_2$ )` : 预览在以  $(x_1, y_1)$  和  $(x_2, y_2)$  为对角顶点的矩形区域内放置商业区的位置。该函数并不实际构建商业区，而是显示商业区确认放置后的位置的可视化表示。
- `try_place_industrial_zone ( $x_1, y_1, x_2, y_2$ )` : Previews the placement of a industrial zone within a rectangular

region with diagonal corners at  $(x_1, y_1)$  and  $(x_2, y_2)$ . This function does not actually construct the industrial zone, but rather displays a visual representation of where the industrial zone would be placed if confirmed.

- `try_place_industrial_zone ( $x_1, y_1, x_2, y_2$ )` : 预览在以  $(x_1, y_1)$  和  $(x_2, y_2)$  为对角顶点的矩形区域内放置工业区的位置。该函数并不实际构建工业区，而是显示工业区确认放置后的位置的可视化表示。

- `try_de_zone ( $x_1, y_1, x_2, y_2$ )` : The function to remove the zone in the game. The zone must cover the road.
- `try_de_zone ( $x_1, y_1, x_2, y_2$ )` : 游戏中移除区域的函数。该区域必须覆盖道路。
- `confirm_placement()`: The function to confirm the placement and build the object after the `try_place_[object]` function.
- `confirm_placement()`: 在 `try_place_[object]` 函数之后确认放置并建造对象的函数。
- `cancel_placement()`: The function to cancel the placement of the object after the `try_place_[object]` function.
- `cancel_placement()`: 在 `try_place_[object]` 函数之后取消对象放置的函数。

Episodic Memory. Besides the common information to store in the episodic memory. We initialize the memory with the coordinates of the entry and exit of the highway. Then CRADLE is able to extend the roads according to these two points at the beginning. When a road or a facility such as wind turbine, water pumping station, water drain pipe and water pipe is placed on the map, the corresponding coordinates will also be stored in the memory for future development of the city.

情景记忆。除了存储在情景记忆中的常见信息外，我们还用高速公路的入口和出口坐标初始化记忆。然后 CRADLE 能够根据这两个点在初期扩展道路。当道路或风力涡轮机、水泵站、排水管和水管等设施被放置在地图上时，相应的坐标也会存储在记忆中，以便未来城市的发展。

## 42.9 E.5 Case Studies

### 42.10 E.5 案例研究

#### 42.10.1 E.5.1 Failure for Road Building.

#### 42.10.2 E.5.1 道路建设失败。

As shown in Figure 38, sometimes GPT-40 will build a long road, which ends on the top of water. The recorded endpoint of the road is actually the projection of the road on the sea level, resulting in the offset from the projection point and the real endpoint of the road. It leads to the failure of extending the road to the other places. 如图38所示，有时 GPT-40 会建造一条长路，终点位于水面上方。道路的记录终点实际上是道路在海平面上的投影，导致投影点与道路实际终点存在偏差。这导致无法将道路延伸到其他地方。

Figure 38b, 38c, 38d and 38e tells a story that GPT-40 sometimes forgets to confirm the placement (from 38c to 38d) and directly moves to the next step of building the next road (from 38d to 38e), resulting in the disconnection of the roads.

图38b、38c、38d和38e讲述了一个故事：GPT-40 有时会忘记确认放置（从38c到38d），直接进入下一步建造下一条道路（从38d到38e），导致道路断开。

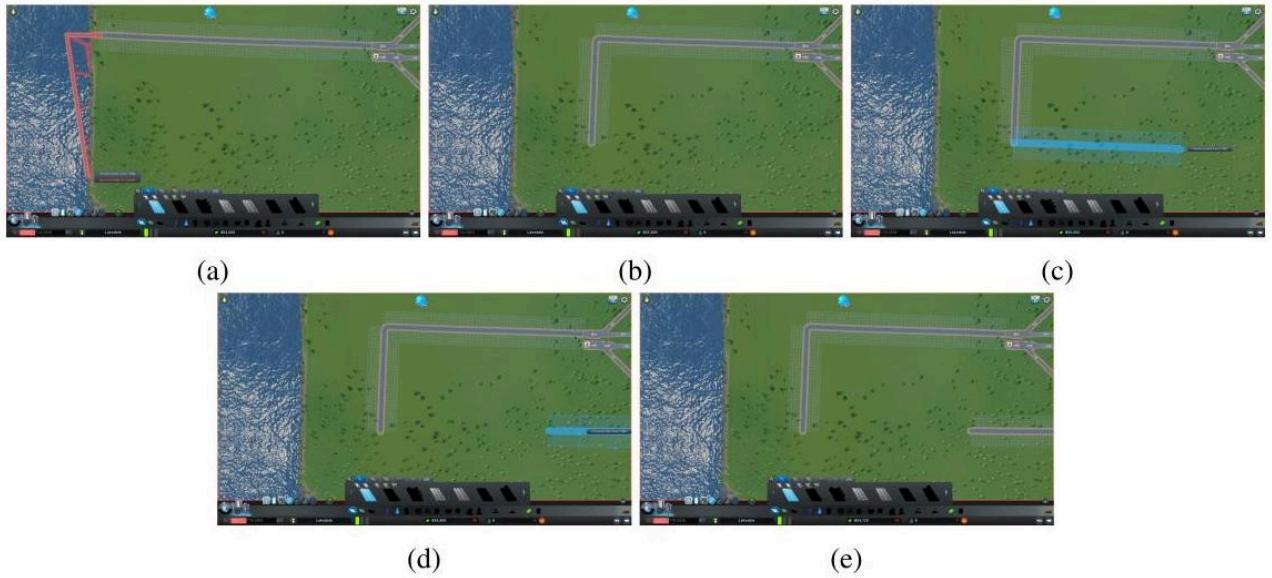


Figure 38: Failure cases of building roads in a closed loop. Figure 38a shows that the road is built over the water and is difficult to continue. Figure 38b, 38c, 38d and 38e tells a story that GPT-40 sometimes forgets to confirm the placement (from 38c to 38d) and directly moves to the next step of building (from 38d to 38e), resulting in the disconnection of the roads.

图38：闭环道路建设失败案例。图38a显示道路建在水面上，难以继续。图38b、38c、38d和38e讲述了GPT-40有时忘记确认放置（从38c到38d），直接进入下一步建造（从38d到38e），导致道路断开。

#### 42.10.3 E.5.2 Failure for Sufficient Water Supply.

#### 42.10.4 E.5.2 供水不足失败。

Figure 39 displays three cases where CRADLE fails to ensure the water supply due to the disconnection of water pipes and the missing water pumping station. All of them can be fixed within three unit operations. As shown in Figure 39b and 39f, we observe a significant increase in the population if these mistakes are fixed, which proves that CRADLE already has the ability to build a reasonable city but some minor adjustments are needed.

图39展示了三种因水管断开和缺少水泵站导致CRADLE供水失败的情况。所有问题均可在三次单元操作内修复。如图39b和39f所示，修复这些错误后人口显著增加，证明CRADLE已具备建设合理城市的能力，但需要一些细微调整。



(a) CRADLE's craftwork I. The upper left corner of the city is experiencing a severe local water shortage since the water pipes there are not connected. **Population: 800+**.



(b) CRADLE's craftwork I with human assistant within three unit operations to develop the idle area in the upper right corner of the city into a residential zone and put two water pipes to ensure all the water pipes connected and cover the whole city. **Population: 1150+**.



(c) CRADLE's craftwork II. The left side of the city a localized area on the right suffers from water shortage because of the water pipes connected issues. **Population: 640+**.



(d) CRADLE's craftwork II with human assistant within three unit operations by selling the redundant water pumping station and the independent water pipe on the right to get some budget and using the budget to get the water pipes connected. **Population: 730+**.



(e) CRADLE's craftwork III. The entire city is experiencing a severe water shortage due to the lack of the water pumping station. **Population: 200+**.



- (f) CRADLE's craftwork III with human assistant within three unit operations to place the water pumping station, lay water pipe on the right side and develop the bottom area with industrial zones. Population: 780+.
- (f) CRADLE 在人类助手协助下的第三阶段工艺，三次单元操作内放置水泵站、铺设右侧水管并在底部区域发展工业区。人口：780+。

Figure 39: Demonstrations of three cities built by CRADLE in zoning view (left), water view (middle) and electricity view (right). Figures 39b, 39d, 39f show the cities with human assistance to address construction issues (shown in red arrow). Populations shown in the figures are close to but not exactly the maximal population since they are changed dynamically.

图39: CRADLE 建造的三个城市示意图，分别为分区视图（左）、水资源视图（中）和电力视图（右）。图39b、39d、39f显示了人类协助解决建设问题（红色箭头所示）的城市。图中显示的人口接近但不完全是最大人口，因为人口数动态变化。

## 43 F Software Applications

### 44 F 软件应用

#### 44.1 F.1 Selected Software Applications

#### 44.2 F.1 选定的软件应用

Besides targeting complex digital games, CRADLE also includes an initial benchmark task set across diverse software applications. The selected applications include Chrome, Outlook, Feishu, CapCut, and Meitu. These applications cover popular applications for daily tasks in different usage categories, such as web browsing, communication, work, and media manipulation. Table 12 shows the exact application versions benchmarked in this paper. Five distinct tasks were designed for each application to represent their target domains and explore the difficulties posed to LMM-based agents and analyze their limitations. Figure 3 shows an overview of all tasks across applications and Tables 13 and 14 detail each task.

除了针对复杂的数字游戏，CRADLE还包含了一个涵盖多样软件应用的初始基准任务集。所选应用包括Chrome、Outlook、飞书（Feishu）、剪映（CapCut）和美图（Meitu）。这些应用覆盖了不同使用类别中日常任务的热门应用，如网页浏览、通信、办公和媒体处理。表12展示了本文基准测试中使用的具体应用版本。为每个应用设计了五个不同的任务，以代表其目标领域，探讨基于大规模语言模型（LMM）代理面临的挑战并分析其局限性。图3展示了所有应用任务的概览，表13和表14详细说明了每个任务。

Chrome and Outlook were selected as common representatives for web browsing and e-mail, with well-known functionality and UI design. CapCut and Meitu are two popular media editing applications for video/image editing with their own interaction styles. Lastly, Feishu (also known as Lark) is an office collaboration and productivity application, which includes messaging, calendar/meetings, and approval workflows. It represents a complex business application that doesn't strictly follow OS-specific UI guidelines. To the best of our knowledge, this is the first agent targeting applications like CapCut, Meitu, and Feishu.

Chrome和Outlook被选为网页浏览和电子邮件的常见代表，具有广为人知的功能和用户界面设计。剪映（CapCut）和美图（Meitu）是两款流行的媒体编辑应用，分别用于视频/图像编辑，拥有各自的交互风格。最后，飞书（Feishu，也称Lark）是一款办公协作和生产力应用，包含消息、日历/会议和审批流程。它代表了一类不严格遵循操作系统特定用户界面指南的复杂商务应用。据我们所知，这是首个针对剪映、美图和飞书等应用的代理。

#### 44.2.1 F.1.1 Brief Descriptions

#### 44.2.2 F.1.1 简要描述

Software	Version
Chrome	125.0.6422.142
Outlook	1.2024.529.200
CapCut	4.0.0
Meitu	7.5.6.1
Feishu	7.19.5

软件	版本
Chrome (谷歌浏览器)	125.0.6422.142
Outlook (微软邮箱)	1.2024.529.200
CapCut (剪映)	4.0.0
美图	7.5.6.1
飞书	7.19.5

Table 12: Exact software versions utilized in the described experiments. Similar versions should behave similarly.

表12：所述实验中使用的确切软件版本。相似版本应具有相似的表现。

Chrome is a web browser developed by Google. It allows users to access and utilize online resources through activities such as browsing websites, streaming videos, and using web applications. Additionally, users can customize their browsing experience with various extensions, manage bookmarks and passwords, and synchronize their data across multiple devices for seamless access.

Chrome是由谷歌开发的网页浏览器。它允许用户通过浏览网站、观看视频和使用网页应用等活动访问和利用在线资源。此外，用户可以通过各种扩展程序自定义浏览体验，管理书签和密码，并在多设备间同步数据，实现无缝访问。

Outlook is an application by Microsoft that allows users to manage emails, calendars, contacts, and tasks. It includes tools for communication and scheduling through features such as sending and receiving emails, setting up meetings, and Outlook是一款应用程序，允许用户管理电子邮件、日历、联系人和任务。它包含用于通信和日程安排的工具，如发送和接收邮件、安排会议等功能。

keeping track of appointments. Additionally, users can customize their experience and integrate Outlook with other Microsoft Office applications.

并跟踪预约。此外，用户可以自定义使用体验，并将Outlook与其他Microsoft Office应用程序集成。

CapCut is a popular video editing application developed by ByteDance. It provides easy-to-use editing tools and enables users to create quality videos with a range of advanced features. CapCut offers a set of editing tools, including trimming, cutting, merging, and splitting video clips; the application of various effects, filters, and transitions; as well as adjusting speed, and adding music or text overlays.

CapCut是字节跳动开发的一款流行视频编辑应用。它提供易用的编辑工具，使用户能够利用多种高级功能制作高质量视频。CapCut提供一套编辑工具，包括剪裁、切割、合并和拆分视频片段；应用各种特效、滤镜和转场；调整速度，以及添加音乐或文字覆盖。

Meitu is a photo editing application. It is designed to cater to a broad audience and enables users to enhance and transform their photos with minimal effort. Meitu offers editing tools, including basic adjustments like cropping, rotating, and resizing, as well as advanced features such as beauty retouching, filters, and special effects.

Additionally, Meitu offers a wide range of stickers, frames, and text options to further personalize photos.

美图是一款照片编辑应用，面向广泛用户，帮助用户轻松提升和变换照片。美图提供编辑工具，包括裁剪、旋转和调整大小等基础操作，以及美颜修饰、滤镜和特效等高级功能。此外，美图还提供丰富的贴纸、边框和文字选项，进一步个性化照片。

Feishu, also known as Lark, is a business communication and collaboration platform by ByteDance. It integrates various tools for office workflows and project management. Feishu offers a wide array of functionalities, including instant messaging, video conferencing, file sharing, and collaboration within the app. It also includes an integrated calendar, which helps users schedule and manage meetings and events, and task management tools that allow users to assign and track tasks.

飞书 (Feishu, 也称Lark) 是字节跳动推出的企业通信与协作平台。它整合了多种办公流程和项目管理工具。飞书提供丰富功能，包括即时消息、视频会议、文件共享和应用内协作。还内置日历，帮助用户安排和管理会议及活动，以及任务管理工具，支持任务分配和跟踪。

#### 44.3 F.2 Software Tasks

#### 44.4 F.2 软件任务

For each of the five applications, we selected a set of representative tasks for their respective domains. For example, search, navigation, and settings tasks on Chrome; sending, searching, and deleting emails, plus changing settings on Outlook; basic video and image editing operations on CapCut and Meitu (e.g., adding special effects and creating a collage); and communication and organization

针对这五款应用，我们为各自领域选择了一组代表性任务。例如，Chrome上的搜索、导航和设置任务；Outlook上的发送、搜索和删除邮件，以及更改设置；CapCut和美图上的基础视频和图像编辑操作（如添加特效和制作拼贴）；以及通信和组织相关任务。

Table 13: Task Descriptions for Chrome, Outlook, and CapCut. Difficulty refers to how hard it is for our agent to accomplish the corresponding tasks. Figures 40, 41, and 42 illustrate each task (specific sub-figures marked in parenthesis in the left-most column along with task name).

表13：Chrome、Outlook和CapCut的任务描述。难度指我们的代理完成相应任务的难易程度。图40、41和42展示了每个任务（最左列任务名称旁括号内标注具体子图）。

Software	Description	Difficulty
	Chrome	
Download Paper (Fig. 40a)	Search for an article with a title like \{paper_title\} and download its PDF file.	Hard
Post in Twitter (Fig. 40b)	Post "It's a good day." on my Twitter.	Hard
Open Closed Page (Fig. 40c)	Open the last closed page.	Easy
Go to Profile (Fig. 40d)	Find and navigate to \{person_name\}'s homepage on GitHub.	Medium
Change Mode (Fig. 40e)	Customize Chrome to dark mode.	Medium
	Outlook	
Send New E-mail (Fig. 41a)	Create a new e-mail to \{email_address\} with subject "Hello friend" and send it.	Medium
Empty Junk Folder (Fig. 41b)	Open the junk folder and delete all messages in it, if any.	Medium
Reply to Person (Fig. 41c)	Open an e-mail from \{person_name\} in the inbox. reply to it with "Got it. Thanks.", and click send.	Medium
Find Target E-mail (Fig. 41d)	Find the e-mail whose subject is "Urgent meeting" and open it.	Easy
Setup Forwarding (Fig. 41e)	Set up email forwarding for every email received to go to \{email_address\}.	Medium
	CapCut	
Create Media Project (Fig. 42a)	Create a new project, then import \{video_file_name\} to the media, click the "Audio" button to add music to the timeline, and finally export the video.	Hard
Add Transition (Fig. 42b)	Open the first existing project. Switch to Transitions panel. Drag a transition effect between the two videos, and then export the video.	Medium
Crop by Timestamp (Fig. 42c)	Delete the video frames after five seconds and then before one second in this video, and then export the video.	Medium
Add Sticker (Fig. 42d)	Open the first existing project. Switch to Stickers panel. Drag a sticker of a person's face to the video, and then export the video.	Hard
Crop by Content (Fig. 42e)	Crop the video when the ball enters the goal, and then export the video.	Very hard

软件	描述	难度
	Chrome浏览器	
下载论文（图40a）	搜索标题类似于 $\{\text{paper\_title}\}$ 的文章并下载其PDF文件。	困难
在Twitter发帖（图40b）	在我的Twitter上发布“今天天气真好。”	困难
打开关闭的页面（图40c）	打开最后关闭的页面。	简单
进入个人主页（图40d）	查找并进入 $\{\text{person\_name}\}$ 在GitHub上的主页。	中等
切换模式（图40e）	将Chrome浏览器自定义为暗黑模式。	中等
	Outlook邮箱	
发送新邮件（图41a）	创建一封发给 $\{\text{email\_address}\}$ 的新邮件，主题为“Hello friend”，并发送。	中等
清空垃圾邮件文件夹（图41b）	打开垃圾邮件文件夹，删除其中所有邮件（如有）。	中等
回复某人（图41c）	打开收件箱中 $\{\text{person\_name}\}$ 的邮件，回复“收到，谢谢。”并点击发送。	中等
查找目标邮件（图41d）	查找主题为“紧急会议”的邮件并打开。	简单
设置邮件转发（图41e）	设置所有收到的邮件自动转发到 $\{\text{email\_address}\}$ 。	中等
	CapCut视频编辑	
创建媒体项目（图42a）	新建项目，导入 $\{\text{video\_file\_name}\}$ 视频，点击“音频”按钮添加音乐到时间线，最后导出视频。	困难
添加转场效果（图42b）	打开第一个已有项目，切换到转场面板，将转场效果拖拽到两个视频间，然后导出视频。	中等
按时间戳裁剪（图42c）	删除视频中五秒后和一秒前的画面，然后导出视频。	中等
添加贴纸（图42d）	打开第一个已有项目，切换到贴纸面板，将一个人物脸部贴纸拖到视频上，然后导出视频。	困难
按内容裁剪（图42e）	在球进入球门时裁剪视频，然后导出视频。	非常困难

operations on Feishu. Tables 13 and 14 describe in detail the 25 tasks CRADLE performs and analyzes on the five selected applications; also illustrated in Figures 40, 41, 42, 43, 44, and 3.

Feishu上的操作。表13和表14详细描述了CRADLE在五个选定应用上执行和分析的25个任务；这些任务也在图40、41、42、43、44和3中有所展示。

It is worth noting that we add a special task on CapCut to demonstrate the agent's ability for tool use. In this task, a pre-defined skill uses GPT-40 as a tool for video understanding capabilities. The skill can be selected to answer content-based questions about a video (e.g., "when the ball enters the goal") and the response be used during task completion. This task is illustrated in detail in Figure 51.

值得注意的是，我们在CapCut上添加了一个特殊任务，以展示代理工具使用能力。在该任务中，预定义技能使用GPT-40作为视频理解工具。该技能可被选用来回答基于视频内容的问题（例如，“球何时进入球门”），并在任务完成过程中使用该回答。该任务在图51中有详细说明。

#### 44.5 F.3 Quantitative Evaluation

#### 44.6 F.3 定量评估

We calculate CRADLE's performance over the 25 tasks in the applications set. Each task is executed five times and performance is measured in three metrics: success rate, average number of steps taken by the agent (and variance over the five runs), and efficiency. Efficiency is defined as the ratio between the expected number of steps in a given task and the total number of steps taken by the agent. The expected number of steps per task is calculated by having humans perform each task.

我们计算了CRADLE在应用集中的25个任务上的表现。每个任务执行五次，性能通过三个指标衡量：成功率、代理

执行的平均步骤数（及五次运行的方差）和效率。效率定义为给定任务的预期步骤数与代理实际执行总步骤数的比值。每个任务的预期步骤数由人工执行该任务计算得出。

Table 14: Task Descriptions for: Meitu, and Feishu. Difficulty refers to how hard it is for our agent to accomplish the corresponding tasks. Figures 43, and 44 illustrate each task (specific sub-figures marked in parenthesis in the left-most column along with task name).

表14：Meitu和Feishu的任务描述。难度指代理完成相应任务的难易程度。图43和44展示了每个任务（左侧最列中任务名称旁括号内标注了具体子图）。

Software	Description	Difficulty
Meitu		
Apply Filter (Fig. 43a)	Apply a filter from Meitu to \{picture_file_name\} and save the project.	Easy
Cutout (Fig. 43b)	Cutout a person from \{picture_file_name\} and save the project.	Easy
Add Sticker (Fig. 43c)	Add a flower sticker to \{picture_file_name\} and save the picture.	Middle
Create Collage (Fig. 43d)	Make a collage using 3 pictures and save the project.	Hard
Add Frame (Fig. 43e)	Add a circle-shaped frame to \{picture_file_name\} and save the picture.	Hard
Feishu		
Create Appointment (Fig. 44a)	Create a new appointment in my calendar anytime later today with title "Focus time".	Hard
Message Contact (Fig. 44b)	Please send a "Hi" chat message to \{contact_name\}.	Easy
Send File (Fig. 44c)	Send the AWS bill file at \\$\{pdf\}\_{path\} in a chat with \{contact_name\}.	Hard
Set User Status (Fig. 44d)	Open the user profile menu and set my status to "In meeting".	Medium
Start Video Conference (Fig. 44e)	Create a new meeting and meet now.	Easy
软件	描述	难度
美图		
应用滤镜（图43a）	对\{picture_file_name\}应用美图滤镜并保存项目。	简单
抠图（图43b）	从\{picture_file_name\}中抠出人物并保存项目。	简单
添加贴纸（图43c）	给\{picture_file_name\}添加一朵花贴纸并保存图片。	中等
制作拼贴（图43d）	使用3张图片制作拼贴并保存项目。	困难
添加边框（图43e）	给\{picture_file_name\}添加一个圆形边框并保存图片。	困难
飞书		
创建预约（图44a）	在我的日历中今天稍晚时间创建一个标题为“专注时间”的新预约。	困难
发送消息给联系人（图44b）	请向\{contact_name\}发送一条“Hi”的聊天消息。	简单
发送文件（图44c）	在与\{contact_name\}的聊天中发送AWS账单文件\\$\\{pdf\\}_{path\\}。	困难
设置用户状态（图44d）	打开用户资料菜单，将我的状态设置为“开会中”。	中等
开始视频会议（图44e）	创建一个新会议并立即开始。	简单

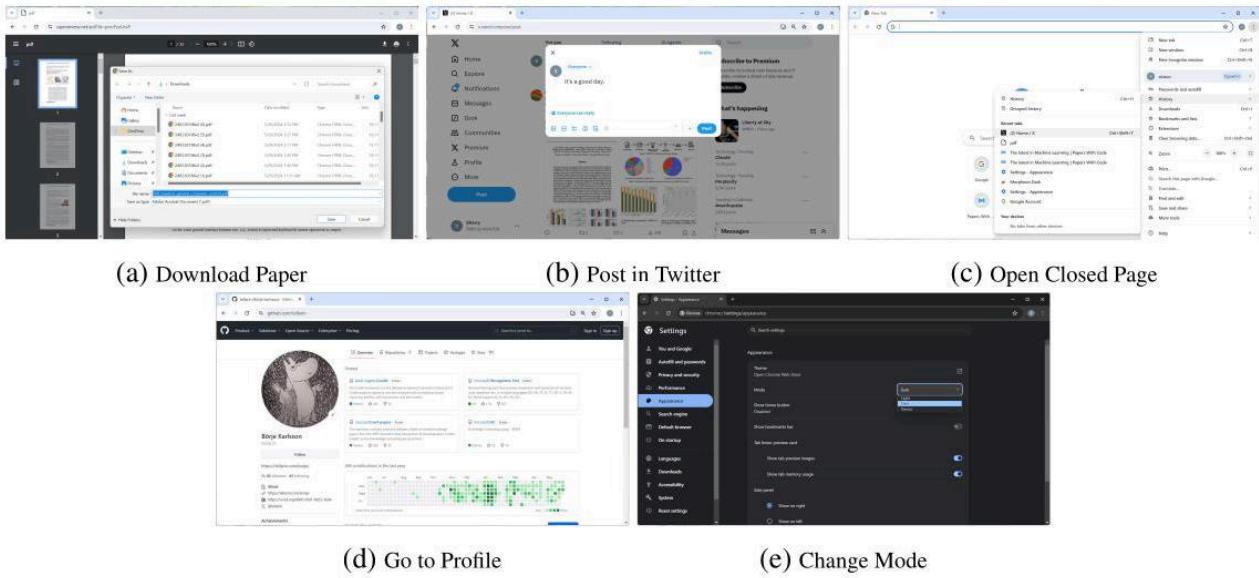


Figure 40: Screenshots of Chrome tasks.

图40：Chrome任务的截图。

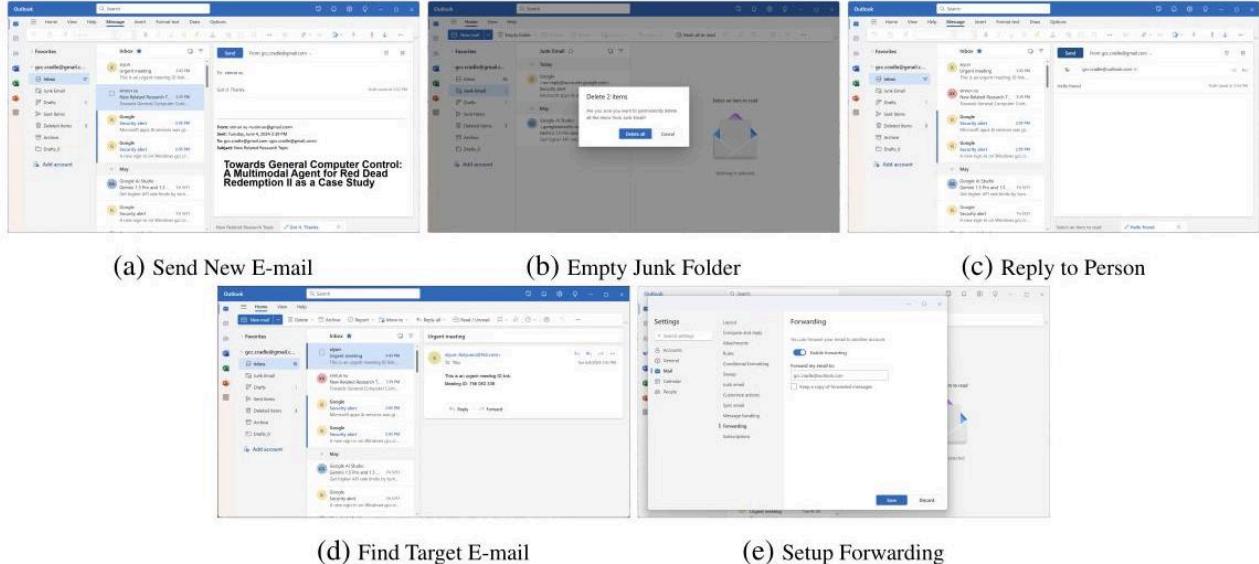


Figure 41: Screenshots of Outlook tasks.

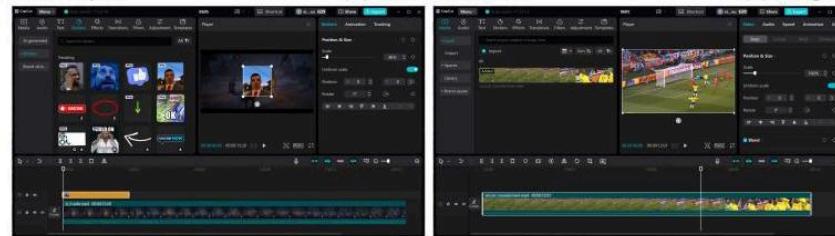
图41：Outlook任务的截图。



(a) Create Media Project

(b) Add Transition

(c) Crop by Timestamp

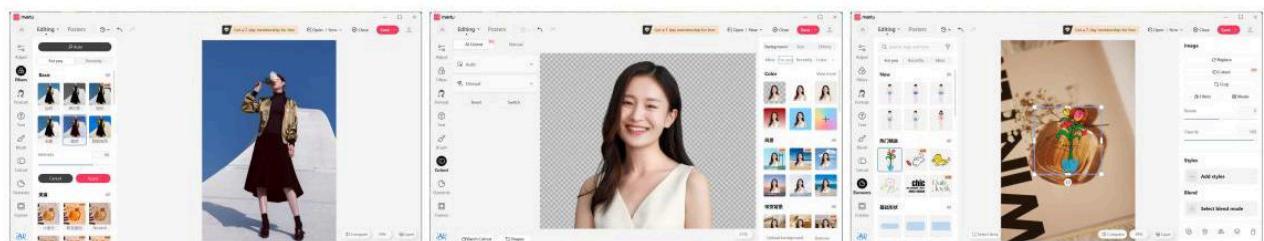


(d) Add Sticker

(e) Crop by Content

Figure 42: Screenshots of CapCut tasks.

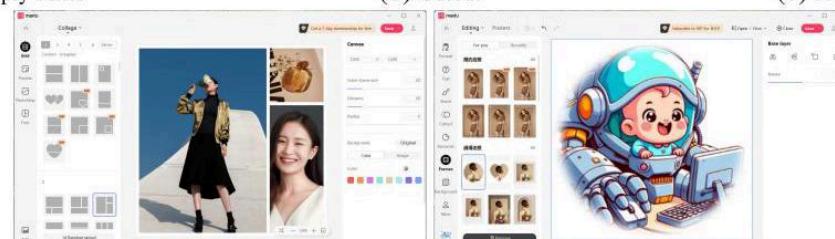
图42：CapCut任务的截图。



(a) Apply Filter

(b) Cutout

(c) Add Sticker



(d) Create Collage

(e) Add Frame

Figure 43: Screenshots of Meitu tasks.

图43：美图任务的截图。

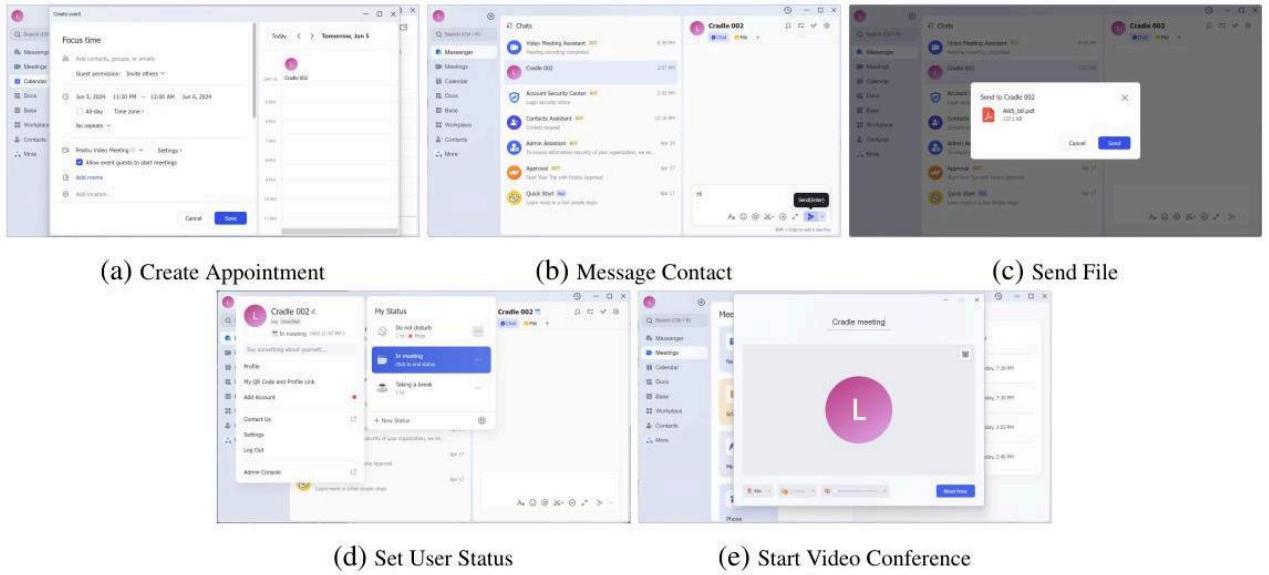


Figure 44: Screenshots of Feishu tasks.

图44：飞书任务的截图。

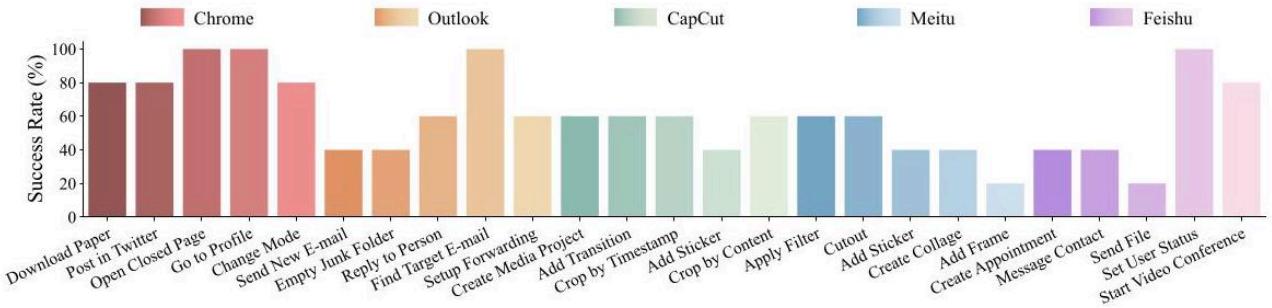


Figure 45: Success rates for tasks in software applications

图45：软件应用中任务的成功率

Table 15 and Figure 45 show the details of the evaluation. CRADLE presents overall good performance over the diverse tasks and applications (compared to Expected Steps, CRADLE achieves an overall efficiency of 50%).

However, performance for certain tasks can vary considerably due to different factors. The main reason for the higher number of task step during agent execution is the frequent incorrect positioning decisions for the mouse, i.e., the backbone model chooses a position of bounding box tag that does not correspond to the UI item described in the model reasoning. We discuss examples of task-specific issues in Sections F.5 and F.6 below.

表15和图45展示了评估的详细情况。CRADLE在多样化的任务和应用中表现整体良好（与预期步骤相比，CRADLE实现了50%的整体效率）。然而，某些任务的表现可能因不同因素而有较大差异。代理执行过程中任务步骤数较多的主要原因是鼠标定位决策频繁错误，即骨干模型选择的边界框标签位置与模型推理中描述的UI项不符。我们在下面的F.5和F.6节中讨论了任务特定问题的示例。

It is worth noting that in Chrome's task 3 ("Open the last closed page"), CRADLE knows how to use the shortcut key directly, calling the key\_press skill directly with the correct keyboard shortcut: 'Ctrl + Shift +T' ,whereas humans typically do not know this.

值得注意的是，在Chrome的任务3（“打开最后关闭的页面”）中，CRADLE知道如何直接使用快捷键，直接调用key\_press技能并使用正确的键盘快捷键：“Ctrl + Shift +T”，而人类通常并不知晓这一点。

To further evaluate the performance of CRADLE in diverse software applications scenarios, we provide quantitative results over OSWorld, a new contemporaneous benchmark with similar characteristics to our settings. More details in Appendix G and overview of the results in Table 16.

为了进一步评估CRADLE在多样软件应用场景中的表现，我们提供了在OSWorld上的定量结果，OSWorld是一个具有与我们设置相似特征的新近基准测试。更多细节见附录G，结果概览见表16。

#### 44.7 F.4 Implementation Details

#### 44.8 F.4 实现细节

The implementation of CRADLE targeting all five software applications follows the GCC setting and framework modules (which include Information Gathering, Self-Reflection, Task Inference, Skill Curation, Action Planning, and Action Execution). Implementation details of the overall framework are described in Appendix A. Therefore, here we emphasize any application-specific differences or customization.

针对所有五个软件应用的CRADLE实现遵循GCC设置和框架模块（包括信息收集、自我反思、任务推断、技能策划、行动规划和行动执行）。整体框架的实现细节描述见附录A。因此，这里重点说明任何应用特定的差异或定制内容。

To apply CRADLE to the target application set described in this appendix, we start with base common prompts, and customize those prompts for specific modules, if necessary, to handle application-specific characteristics. For example, for CapCut we add few-shot examples for Self-Reflection, to let it properly perform success detection, as the application UI by itself is non-standard and sometimes provides little post-action feedback to users, making it harder for the backend model to determine action success.

为了将CRADLE应用于本附录描述的目标应用集，我们从基础通用提示开始，必要时为特定模块定制提示以处理应用特性。例如，对于CapCut，我们为自我反思模块添加了少量示例，以使其能够正确执行成功检测，因为该应用的UI本身非标准且有时对用户的操作后反馈较少，增加了后端模型判断操作成功的难度。

Table 15: Application Software results. Success Rate determines the ratio of successful completions over five runs. Average Steps refers to the number of actions the agent takes to fulfil a task, if successful. Expected Steps represents the number of steps as estimated by humans performing the task. Efficiency represents the ratio between the expected number of steps and the total number of steps taken by the agent.

表15：应用软件结果。成功率表示五次运行中成功完成的比例。平均步骤数指代理成功完成任务所采取的动作数。预期步骤数表示人类执行任务时估计的步骤数。效率表示预期步骤数与代理实际采取总步骤数的比率。

Software	Success Rate	Average Steps	Expected Steps	Efficiency
Chrome	88%	8.74	4.20	48.05%
Download Paper	80%	16.00 ± 5.52	6	37.50%
Post in Twitter	80%	11.75 ± 5.26	7	61.14%
Open Closed Page	100%	1.00 ± 0	3	300.00%
Go to Profile	100%	4.00 ± 0.63	1	25.00%
Change Mode	80%	11.25 ± 4.71	4	35.56%
Outlook	60%	8.25	4	48.48%
Send New E-mail	40%	11.00 ± 4	5	45.45%
Empty Junk Folder	40%	8.50 ± 3.50	3	35.29%
Reply to Person	60%	8.33 ± 4.71	4	48.02%
Find Target E-mail	100%	1.40 ± 0.80	1	71.43%
Setup forwarding	60%	12.00 ± 4.90	7	58.33%
CapCut	56%	10.87	4.80	44.16%
Create Media Project	60%	13.67 ± 5.25	7	51.20%
Add transition	60%	10.67 ± 4.03	4	37.49%
Crop by Timestamp	60%	11.00 ± 5.66	5	45.45%
Add Sticker	40%	12.00 ± 8.00	4	33.33%
Crop by Content	60%	7.00 ± 1.41	4	57.14%
Meitu	44%	12.5	8.00	64%
Apply Filter	60%	14.67 ± 2.36	7	47.72%
Cutout	60%	9.33 ± 1.89	5	53.59%
Add Sticker	40%	9.50 ± 0.50	8	84.21%
Create Collage	40%	16.00 ± 2.00	12	75.00%
Add Frame	20%	13.00 ± 0.00	7	53.85%
Feishu	56%	8.82	4	46.07%
Create Appointment	40%	8.00 ± 1.00	4	50.00%
Message Contact	40%	6.00 ± 1.00	3	50.00%
Send file	20%	11.00 ± 0.00	7	63.64%
Set User Status	100%	14.60 ± 7.50	3	20.55%
Start Video Conference	80%	4.50 ± 2.60	3	46.15%

软件	成功率	平均步骤数	期望步骤数	效率
Chrome浏览器	88%	8.74	4.20	48.05%
下载论文	80%	16.00 ± 5.52	6	37.50%
在Twitter发布	80%	11.75 ± 5.26	7	61.14%
打开已关闭页面	100%	1.00 ± 0	3	300.00%
进入个人资料	100%	4.00 ± 0.63	1	25.00%
切换模式	80%	11.25 ± 4.71	4	35.56%
Outlook邮箱	60%	8.25	4	48.48%
发送新邮件	40%	11.00 ± 4	5	45.45%
清空垃圾邮件文件夹	40%	8.50 ± 3.50	3	35.29%
回复联系人	60%	8.33 ± 4.71	4	48.02%
查找目标邮件	100%	1.40 ± 0.80	1	71.43%
设置邮件转发	60%	12.00 ± 4.90	7	58.33%
CapCut视频剪辑	56%	10.87	4.80	44.16%
创建媒体项目	60%	13.67 ± 5.25	7	51.20%
添加转场效果	60%	10.67 ± 4.03	4	37.49%
按时间截裁剪	60%	11.00 ± 5.66	5	45.45%
添加贴纸	40%	12.00 ± 8.00	4	33.33%
按内容裁剪	60%	7.00 ± 1.41	4	57.14%
美图秀秀	44%	12.5	8.00	64%
应用滤镜	60%	14.67 ± 2.36	7	47.72%
抠图	60%	9.33 ± 1.89	5	53.59%
添加贴纸	40%	9.50 ± 0.50	8	84.21%
创建拼贴	40%	16.00 ± 2.00	12	75.00%
添加边框	20%	13.00 ± 0.00	7	53.85%
飞书	56%	8.82	4	46.07%
创建预约	40%	8.00 ± 1.00	4	50.00%
发送消息给联系人	40%	6.00 ± 1.00	3	50.00%
发送文件	20%	11.00 ± 0.00	7	63.64%
设置用户状态	100%	14.60 ± 7.50	3	20.55%
开始视频会议	80%	4.50 ± 2.60	3	46.15%

Information Gathering. Noticeably, GPT-4o presents the same limitations in both spatial reasoning (e.g., confusing up/down, left/right) and image understanding identifying specific UI items or the state of the forefront GUI, across all applications.

信息收集。显著的是，GPT-4o在空间推理（例如，混淆上下、左右方向）和图像理解（识别特定UI元素或前端GUI状态）方面，在所有应用中均表现出相同的局限性。

To help mitigate such issues, we perform augmentation on the captured screenshots similarly to the Set-of-Mark (SoM) approach [68], by only utilizing SAM [29] to generate potential UI items bounding boxes and assign them numerical tags. Our SoM-like augmentation differs from recent agent-related work (e.g., [66, 73]), which use OS-specific APIs to draw ground-truth bounding boxes for interactable elements (plus UI structure info, like types and element tree) to the results, while CRADLE relies only on image input and the segmentation output as augmentation. To make this distinction explicit, we call our augmentation approach SAM2SOM<sup>11</sup>. Figure 49 illustrates the difference. While our approach produces many more potential bounding boxes, it is more general by relying only on a screenshot (or video frame).

为帮助缓解此类问题，我们对捕获的截图进行增强，方法类似于Set-of-Mark (SoM) 方法[68]，仅利用SAM[29]生成潜在UI元素的边界框并分配数字标签。我们的SoM类增强不同于近期的代理相关工作（如[66, 73]），后者使用操作

系统特定API绘制可交互元素的真实边界框（以及UI结构信息，如类型和元素树）作为结果，而CRADLE仅依赖图像输入和分割输出作为增强。为明确区分，我们将此增强方法称为SAM2SOM<sup>11</sup>。图49展示了差异。虽然我们的方法生成了更多潜在边界框，但其更通用，仅依赖截图（或视频帧）。

<sup>11</sup> We do not claim the method itself as a core contribution. SAM2SOM is used to illustrate a possible extra capability of the backend model, as mitigation for current spatial reasoning issues.

<sup>11</sup> 我们不将该方法本身视为核心贡献。SAM2SOM用于展示后端模型可能具备的额外能力，以缓解当前的空间推理问题。

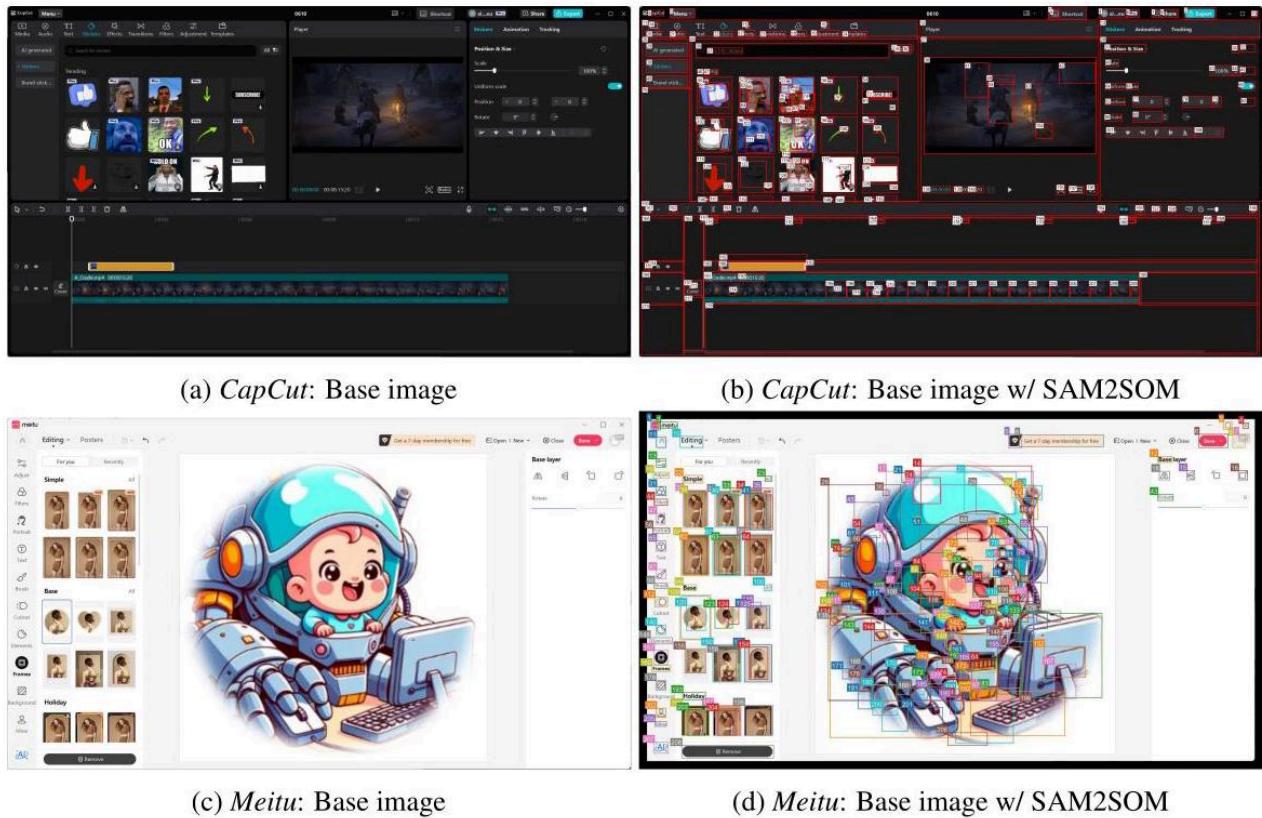


Figure 47: Image examples of the two SAM2SOM augmentation styles. As CapCut's UI (top row) has very dark background, we utilize single-color borders with IDs in black text over white background, placed within the bounding box area. Other application software and OSWorld use the "standard" SAM2SOM multi-color style, as shown for Meitu (bottom row).

图47：两种SAM2SOM增强样式的图像示例。由于CapCut的UI（顶行）背景非常暗，我们采用单色边框，标签为黑色文字白色背景，置于边界框内。其他应用软件和OSWorld使用“标准”SAM2SOM多色样式，如美图（底行）所示。

To ensure all bounding box labels are consistently positioned, CRADLE's SAM2SOM implements two rendering styles, as shown in Figure 47 first and second rows. In the standard style, we pad the SAM2SOM-enhanced image when showing the label IDs in the upper left corner of the bounding boxes (to prevent labels from hiding the contents of small areas), so no numerical label ID is drawn outside the image area). In the uniform style, all bounding boxes utilize single-color borders with labels in black text over white background, placed within the

bounding box area (top left corner).

为确保所有边界框标签位置一致，CRADLE的SAM2SOM实现了两种渲染样式，如图47第一和第二行所示。在标准样式中，显示标签ID于边界框左上角时，会对SAM2SOM增强图像进行填充（防止标签遮挡小区域内容），因此无数字标签ID绘制在图像外部。在统一样式中，所有边界框均采用单色边框，标签为黑色文字白色背景，置于边界框内左上角。

Moreover, in specific situations we may still need to refine SAM2SOM's output further. For example, in the Feishu case, we observe that watermarks generated by the software affect the segmentation negatively, complicating GPT-4o's selection of the correct bounding boxes to interact with. Therefore, we implement a simple filtering method for such watermarks. This filter is enabled only in the Feishu benchmark and, as shown in Figure 48, can greatly reduce the number of unnecessary bounding boxes (from 216 to 166, in this example).

此外，在特定情况下，我们可能仍需进一步优化SAM2SOM的输出。例如，在飞书案例中，软件生成的水印会负面影响分割，增加GPT-4o选择正确交互边界框的难度。因此，我们实现了针对该水印的简单过滤方法。该过滤器仅在飞书基准中启用，如图48所示，可显著减少不必要的边界框数量（本例中从216减少到166）。

In addition to using the SAM2SOM method for image augmentation, we also redraw the mouse pointer not present in captured screenshots in a more prominent magenta color based on its screen position, to emphasize both its presence and position for image understanding (e.g., Figure 46). The augmentation process in Information Gathering can then result in four versions of a screenshot: a) base image, b) SAM2SOM image, c) base image with mouse pointer, and d) SAM2SOM image with mouse pointer.

除了使用SAM2SOM方法进行图像增强外，我们还根据鼠标指针在屏幕上的位置，将截图中未出现的鼠标指针以更显眼的品红色重新绘制，以突出其存在和位置，辅助图像理解（例如图46）。信息收集中的增强过程可生成四种截图版本：a) 基础图像，b) SAM2SOM图像，c) 带鼠标指针的基础图像，d) 带鼠标指针的SAM2SOM图像。

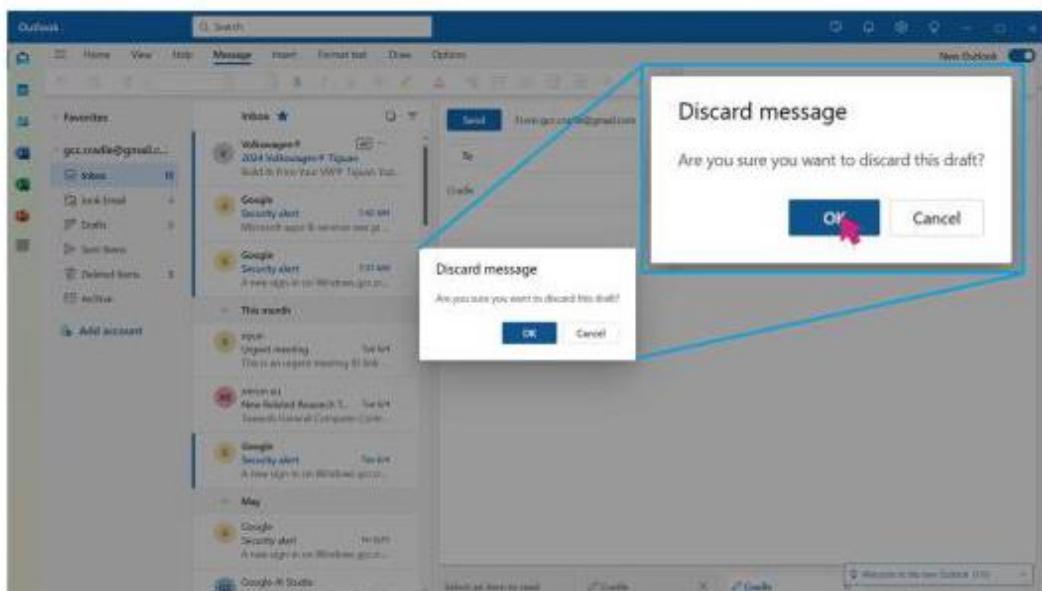


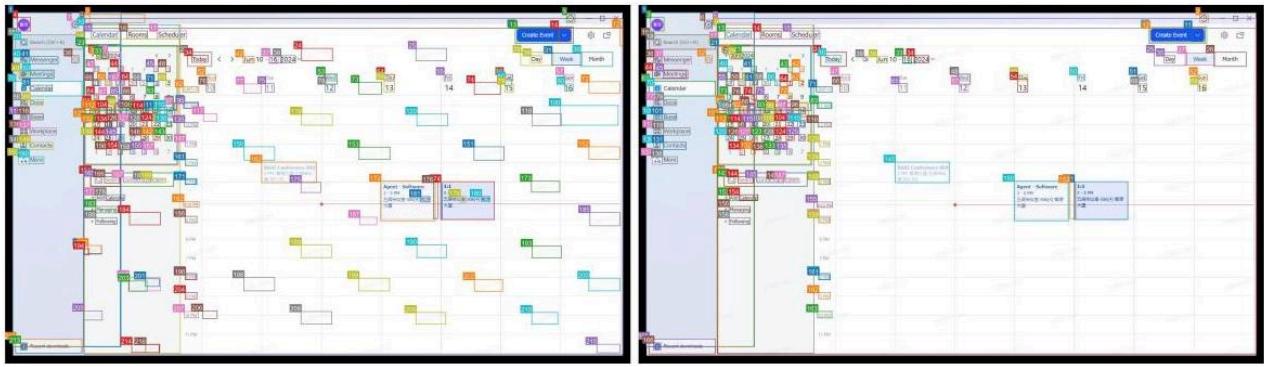
Figure 46: Sample augmented image w/ drawn mouse pointer. Zoom overlay shows the image difference.

图46：带绘制鼠标指针的示例增强图像。放大叠加显示图像差异。

Self-Reflection. As the applications in the software set are much less dynamic than complex games, there is no need to send multiple video

自我反思。由于软件集中的应用远不如复杂游戏动态，因此无需向自我反思模块发送多帧视频

frames to Self-Reflection. For the software applications, pre- and post-action screenshot usually suffice, i.e., one image before and one image after an action is executed. Digital games often have frames. 对于软件应用，动作前后截图通常足够，即动作执行前后各一张图像。数字游戏通常具有



(a) SAM2SOM image w/ watermarks

(b) SAM2SOM image w/o watermarks

Figure 48: Examples of filtering watermark in Feishu. The number of labels is greatly reduced from 216 to 166.  
图48：飞书中水印过滤示例。标签数量从216大幅减少至166。

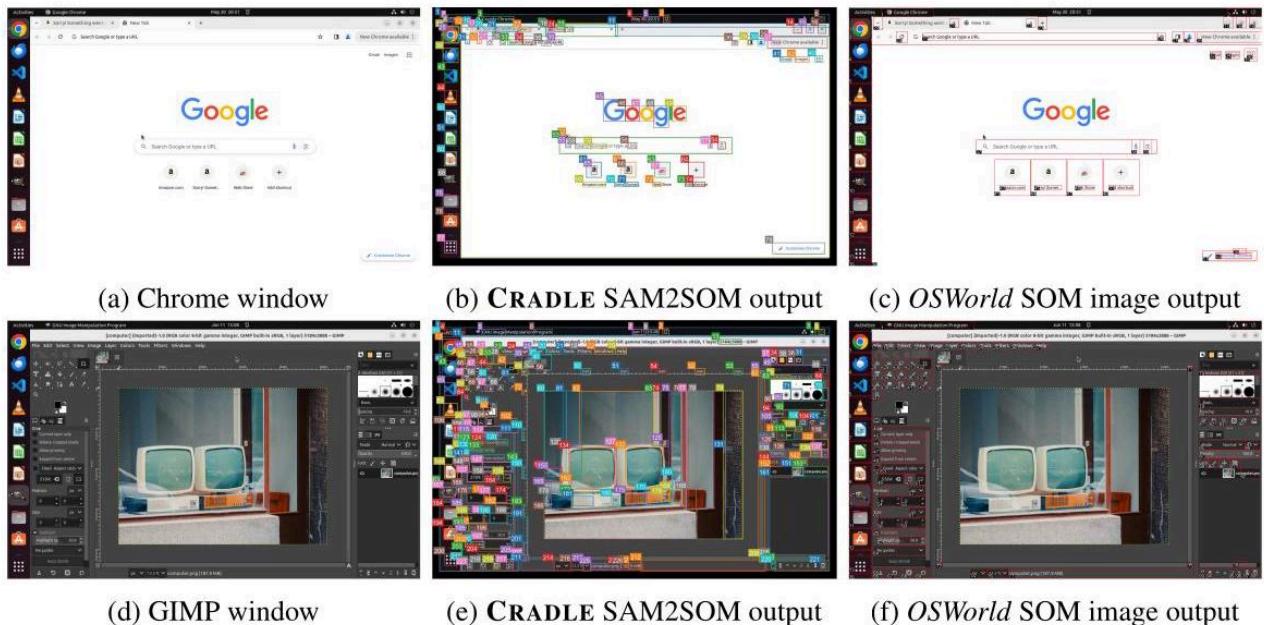


Figure 49: Comparison of CRADLE's visual-only SAM2SOM and OSWorld's API-based SOM image results.

Chrome: 78 vs. 53 bounding boxes; GIMP: 227 vs. 98 bounding boxes.

图49：CRADLE仅视觉SAM2SOM与OSWorld基于API的SOM图像结果对比。Chrome: 78 vs. 53个边界框；  
GIMP: 227 vs. 98个边界框。

continuous and dynamic environments that require multiple frames to properly capture the full context and thus help the backbone LMMs understand what happened. In contrast, software operations are typically more discrete and static, where the state before and after an action provides sufficient information for most analysis.

连续且动态的环境，需多帧捕捉完整上下文，帮助骨干大规模多模态模型（LMMs）理解发生了什么。相比之下，软件操作通常更离散和静态，动作前后状态即可为大多数分析提供充分信息。

Nonetheless, we find that irrespective of images used, GPT-40 sometimes can have difficulty determining the success of certain tasks. For example, when downloading a file on Chrome, after either pressing 'Ctrl + S', or using a 'Save' menu, the agent must also press 'Enter' or click the 'Save' button to complete the task. However, GPT-40 often assumes the task is complete when the dialog opens and before this final step. Similar cases of incorrect conclusion happen when an action correctly closes a new panel or dialog. To address this category of issues, we add mandatory reasoning rules in the prompt for the Self-Reflection module to help mitigate such mistakes. If for

specific applications this still remains an issue, we can use few-shot image examples to reinforce how the backend model should correctly judge success.

尽管如此，我们发现无论使用何种图像，GPT-4o 有时在判断某些任务是否成功时仍会遇到困难。例如，在 Chrome 浏览器下载文件时，无论是按下“Ctrl + S”还是使用“保存”菜单，代理还必须按“Enter”键或点击“保存”按钮才能完成任务。然而，GPT-4o 经常在对话框打开且未执行最后一步时就假设任务已完成。当某个操作正确关闭新面板或对话框时，也会出现类似的错误结论。为了解决这类问题，我们在自我反思模块的提示中添加了强制推理规则，以帮助减少此类错误。如果在特定应用中问题依然存在，我们可以使用少量示例图像来强化后端模型如何正确判断任务成功。

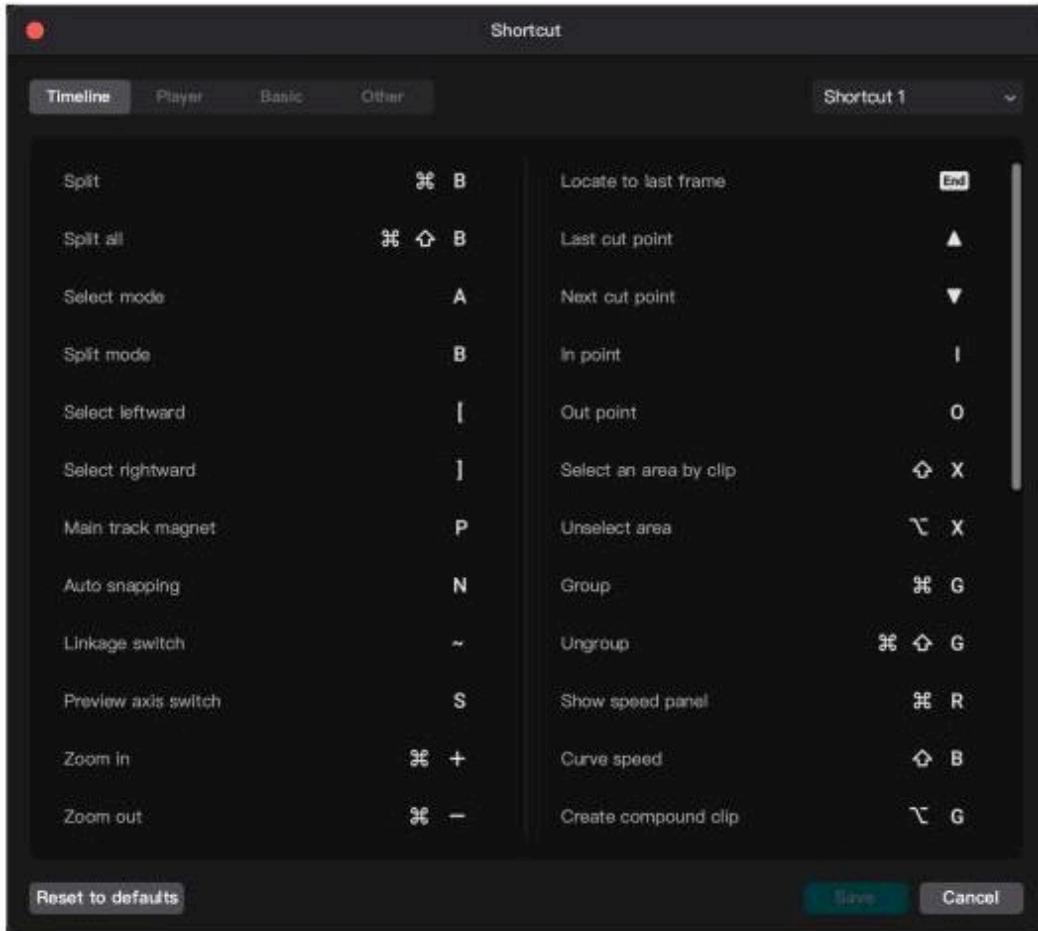


Figure 50: Shortcuts screen in CapCut.

图50：CapCut中的快捷键界面。

**Skill Curation.** In software tasks, direct skill generation was not necessary, as UI operations generally map closely to specific mouse or keyboard actions, making them more straightforward. In contrast, digital game environments involve continuous interactions and decision-making, raising new previously undiscovered information, and requiring the development of new skills to handle novel scenarios and adapt to changing contexts.

技能策划。在软件任务中，通常不需要直接生成技能，因为UI操作通常与特定的鼠标或键盘动作紧密对应，使得操作更为直接。相比之下，数字游戏环境涉及持续的交互和决策，带来新的未曾发现的信息，并需要开发新技能以应对新场景和适应变化的环境。

However, we do add some additional predefined skills, on a per-application basis, for specific knowledge like less-widely known keyboard shortcuts which could be learnt from the application. For example, CapCut's shortcuts screen, shown in Figure 50, or toolbar/icon processing output similarly to the process described for Cities: Skylines. Moreover, we also intro-

然而，我们确实为每个应用添加了一些额外的预定义技能，用于特定知识，如应用中可学习的较少为人知的键盘快捷

键。例如，图50所示的CapCut快捷键界面，或类似于《城市：天际线》（Cities: Skylines）中描述的工具栏/图标处理输出。此外，我们还引入了

duce pre-defined complex skills to demonstrate CRADLE's capability to leverage tools into novel functionality, such as using GPT-4o as a tool to extract information from a video to complete task 5 in CapCut.

预定义的复杂技能，以展示CRADLE利用工具实现新功能的能力，例如使用GPT-4o作为工具从视频中提取信息以完成CapCut中的任务5。

When dealing with shortcuts, e.g., as alternatives to mouse operations, it may be the case that specific shortcuts require "calibration". For example, using the keyboard to navigate the timeline in CapCut (as seen in the bottom area of Figure 47b) requires mapping the keyboard shortcut ('Alt + arrow keys') to pixels or time, which we perform a priori and use the mapping in the pre-defined skill go\_to\_timestamp(seconds).

在处理快捷键时，例如作为鼠标操作的替代方案，可能需要对特定快捷键进行“校准”。例如，使用键盘导航CapCut时间线（见图47b底部区域）需要将键盘快捷键（“Alt + 方向键”）映射到像素或时间，我们事先完成此映射，并在预定义技能go\_to\_timestamp(seconds)中使用该映射。

Task Inference. During the execution of an application task, we let GPT-40 decompose the execution strategy for the next step based on the overall task description and the subtask description. If the previous task decomposition is found to be unreasonable, a new decomposition plan should be proposed and this is evaluated at each iteration round.

任务推断。在执行应用任务时，我们让GPT-4o根据整体任务描述和子任务描述分解下一步的执行策略。如果发现之前的任务分解不合理，应提出新的分解方案，并在每轮迭代中进行评估。

Action Planning. To enable usage of SAM2SOM, for Action Planning, we insert new mouse skills, which mirror existing coordinates-based mouse skills (i.e., that use x,y coordinates), but take a bounding box numerical label as an argument.

动作规划。为了支持SAM2SOM的使用，在动作规划中我们插入了新的鼠标技能，这些技能与现有基于坐标的鼠标技能（即使用x,y坐标）相对应，但以边界框的数字标签作为参数。

Furthermore, unlike in game playing, which focuses on performing one action per turn, when manipulating software CRADLE can be configured to perform two actions in sequence and thus lower interaction frequency requirements to the backend model. We find that GPT4-o can usually correctly output two-step compound actions. For example, when performing a search in the browser, it can typically output two consecutive action steps, e.g., type\_text(text='{user\_query}'), followed by the required press\_key(key='enter').

此外，与游戏中每回合执行单一动作不同，操作软件时CRADLE可以配置为连续执行两个动作，从而降低对后端模型的交互频率要求。我们发现GPT-4o通常能正确输出两步复合动作。例如，在浏览器中执行搜索时，通常能输出连续的两个动作步骤，如type\_text(text='{user\_query}')，随后是必需的press\_key(key='enter')。

Action Execution. While atomic and composite skills can involve complex operations, Action Execution happens over the regular CRADLE action space, as shown in Table 7. For example, during Action Execution, a post-processing step converts the bounding box calls into regular mouse actions, using the centroid of a given bounding box as its coordinates for regular mouse operations.

动作执行。尽管原子技能和复合技能可能涉及复杂操作，动作执行仍在常规CRADLE动作空间内进行，如表7所示。例如，在动作执行过程中，后处理步骤将边界框调用转换为常规鼠标操作，使用给定边界框的中心点作为常规鼠标操作的坐标。

Tool usage, like calling GPT-40 separately to analyze the contents of a media file, is not considered as an action, as tools do not operate on the environment, only as code steps inside a composite skill.

工具使用，如单独调用GPT-4o分析媒体文件内容，不被视为动作，因为工具不直接作用于环境，仅作为复合技能内的代码步骤。

## 44.9 F.5 Case Studies

### 44.10 F.5 案例研究

#### 44.10.1 F.5.1 Task Hardness

##### 44.10.2 F.5.1 任务难度

It is well known that the difficulty of task completion can vary widely between humans and agents. The results in Table 15 help illustrate some such cases. While many application operation issues may be attributed to UI variety or non-conformity, that is not necessarily the main source of task hardness (i.e., how unexpectedly complex performing an operation is).

众所周知，任务完成的难度在人类和代理之间差异很大。表15中的结果有助于说明一些此类情况。虽然许多应用操作问题可能归因于UI多样性或不一致性，但这不一定是任务难度的主要来源（即执行操作的复杂程度出乎意料）。

Here we use Outlook, a well-known e-mail client, as a case study to discuss how different factors affect CRADLE task completion in real-world application situations (the exact version used is listed in Table 12). Taking task 1 ("Create a new e-mail to {email\_address} with the subject 'Hello friend' and send it.") as an example, a success rate of 40% and efficiency of 45.45% may seem lower than expected.

这里我们以知名电子邮件客户端Outlook为例，讨论不同因素如何影响CRADLE在真实应用场景中的任务完成情况（具体版本见表12）。以任务1（“创建一封主题为‘Hello friend’并发送至{email\_address}的新邮件。”）为例，40%的成功率和45.45%的效率可能显得低于预期。

Such a task could be reasonably broken down into steps like: a) Create new e-mail, b) Add recipient, c) Write title, and d) Send e-mail. And the Task Inference module performs such decomposition consistently. However, Action Planning needs to define specific actionable operations with mouse and keyboard to execute each step.

这样一个任务可以合理地分解为以下步骤：a) 创建新邮件，b) 添加收件人，c) 编写标题，d) 发送邮件。任务推断模块能够稳定地执行这种分解。然而，动作规划需要定义具体的鼠标和键盘操作来执行每一步。

Firstly, CRADLE needs to decide based on the knowledge and visual understanding capabilities available to it to either use a known keyboard shortcut (e.g., 'Ctrl + N') or to click at the "New mail" button. In our experiments, CRADLE tends to chose clicking on the button, which is then affected by the previously discussed issues that led to the integration of SAM2SOM into the framework. Issues in spatial reasoning issues or icon/image understanding may cause a few incorrect click attempts.

首先，CRADLE需要基于其可用的知识和视觉理解能力决定是使用已知的键盘快捷键（例如“Ctrl + N”）还是点击“新建邮件”按钮。在我们的实验中，CRADLE倾向于选择点击按钮，这就受到之前讨论过的问题影响，促使将SAM2SOM集成到框架中。空间推理或图标/图像理解的问题可能导致几次错误的点击尝试。

Adding the recipient to the e-mail requires typing an address at the appropriate location, i.e., the typical "To" field. This can be accomplished in multiple ways, mainly by typing the address on the UI next to the "To" item or choosing a pre-existing contact.

向邮件中添加收件人需要在合适的位置输入地址，即典型的“收件人”字段。这可以通过多种方式完成，主要是直接在“收件人”项旁的界面上输入地址，或选择已有联系人。

Clicking on the "To" button triggers the UI to search and select a pre-existing contact e-mail address (with no option of adding a new contact entry, which requires first accessing the "Contacts" menu, outside of "Mail").

Moreover, the UI interaction sequence to select an existing contact can be unintuitive even to experienced users, requiring a minimum of four steps, at each step offering multiple UI options that go away from contact selection. Attempting this flow usually leads CRADLE to exceed the maximum number of allowed step as it gets confused by the UI design.

点击“收件人”按钮会触发界面搜索并选择已有的联系人邮箱地址（没有添加新联系人条目的选项，添加新联系人需要先进入“联系人”菜单，位于“邮件”之外）。此外，选择已有联系人的界面交互流程即使对有经验的用户来说也可能不

直观，至少需要四个步骤，每一步都有多个界面选项，且这些选项会偏离联系人选择。尝试此流程通常导致CRADLE超出允许的最大步骤数，因为它被界面设计弄得困惑。

Nonetheless, choosing the simpler alternative of typing the e-mail address (assuming the correct text field is selected) triggers assistive UI pop-ups (as shown in Figure 52), which lead GPT-4o to falsely conclude the e-mail address is either already typed at the correct location or that it is duplicated and needs to be edited/removed. Furthermore, the pop-ups partially hide the subject area, making it harder for CRADLE to choose the next UI item to interact with for the next task step.

尽管如此，选择更简单的输入邮箱地址的方式（假设选中了正确的文本框）会触发辅助界面弹窗（如图52所示），这导致GPT-4o错误地判断邮箱地址要么已正确输入，要么是重复的，需要编辑或删除。此外，弹窗部分遮挡了主题区域，使CRADLE更难选择下一步要交互的界面元素。

Similar issues with positioning and correctly identifying the typed subject text can also occur, but at a much smaller frequency.

类似的定位和正确识别已输入主题文本的问题也会发生，但频率要低得多。

Lastly, completing the task and sending the e-mail requires step similar to creating a new message. But determining send success requires additional attention/reflection as not all cases of the "Send mail" interface disappearing indicate a successful send (e.g., clicking on an unrelated e-mail on the Inbox or closing the current window pop-up). 最后，完成任务并发送邮件需要类似于创建新邮件的步骤。但判断发送是否成功需要额外关注和反思，因为“发送邮件”界面消失并不总是表示发送成功（例如点击收件箱中无关的邮件或关闭当前弹窗）。

The Self-Reflection module plays a key role in moving task completion forward by detecting failed attempts at executing each sub-task and providing rationale for failures, even if Information Gathering and Action Planning make repeated mistakes. Such feedback from Self-Reflection allows Action Planning to tune its process and move ahead.

自我反思模块在推动任务完成中起关键作用，它能检测每个子任务执行失败的尝试并提供失败原因，即使信息收集和动作规划反复出错。来自自我反思的反馈使动作规划能够调整流程并继续前进。

#### 44.10.3 F.5.2 Tool Use in CapCut

#### 44.10.4 F.5.2 CapCut中的工具使用

Some general computer control tasks may require additional capabilities during execution preparation that can benefit from external tools to enhance agent abilities.

一些通用的计算机控制任务在执行准备阶段可能需要额外能力，这些能力可以通过外部工具提升代理的能力。

When performing video editing, like in CapCut, a user may need to determine the precise frames to operate on based on video content. For such scenarios, CRADLE can easily leverage tool-using skills, like the LMM's ability to understand actions in a sequence of video frames, enabling it to comprehend video content and identify the exact frames for editing.

在进行视频编辑时，如在CapCut中，用户可能需要根据视频内容确定精确的帧进行操作。对于此类场景，CRADLE可以轻松利用工具使用技能，比如大型多模态模型（LMM）理解视频帧序列中动作的能力，从而理解视频内容并识别具体的编辑帧。

We exemplify such tasks with task 5 ("Crop the video when the ball enters the goal, and then export the video") for CapCut, as illustrated in Figure 51. This means our agent can effectively execute tool usage to find the specific frame where "the ball enters the goal". After the first round of Task Inference, CRADLE decomposes the task into three subtasks: 1. Identify the exact frame, 2. Crop the video, and 3. Export the video. Action Planning can then plan to execute 'get\_information\_from\_video(event)' from our curated skills and generate "ball enters the goal" as its required argument for execution.

我们以任务5（“当球进入球门时裁剪视频，然后导出视频”）为例，如图51所示。这意味着我们的代理能够有效执行工具使用，找到“球进入球门”的具体帧。经过第一轮任务推断，CRADLE将任务分解为三个子任务：1. 确定具体

帧, 2. 裁剪视频, 3. 导出视频。动作规划随后可以计划执行我们策划技能中的“get\_information\_from\_video(event)”, 并生成“球进入球门”作为执行所需参数。

In this skill, we input a frame set of the video at 1 fps to identify the specific frame where the event occurs. The response is then recorded in Episodic Memory to ensure that subsequent operations can accurately utilize it and target the moment when the action occurs. Across subsequent iterations, CRADLE can then correctly plan and execute the remaining necessary actions for task completion: 'go\_to\_timestamp(seconds=8)', 'delete\_right()', and 'export\_project()'.

在此技能中, 我们以1帧每秒的帧集输入视频, 以识别事件发生的具体帧。响应结果被记录在情景记忆中, 确保后续操作能准确利用该信息, 定位动作发生的时刻。在后续迭代中, CRADLE能够正确规划并执行完成任务所需的剩余操作: ‘go\_to\_timestamp(seconds=8)’, ‘delete\_right()’, 和 ‘export\_project()’。

We have integrated few-shot learning into Self-Reflection to ensure CRADLE recognizes that following export\_project(), the expected screen is the CapCut application main window. This information allows it to verify the successful execution of the task, leading to success detection for the overall task.

我们已将少量示例学习集成到自我反思模块, 确保CRADLE认识到执行完‘export\_project()’后, 预期界面是CapCut应用主窗口。此信息使其能够验证任务执行成功, 从而实现整体任务的成功检测。

### Task Description: Crop the video when the ball enters the goal, and then export the video.



Figure 51: Showcase of Task 5 ("Crop the video when the ball enters the goal, and then export the new video") in CapCut.

图51: CapCut中任务5 (“当球进入球门时裁剪视频, 然后导出新视频”) 的展示。

#### 44.11 F.6 Limitations of GPT-4o

#### 44.12 F.6 GPT-4o的局限性

Besides the previously discussed limitations of GPT-4o, it is important to highlight a couple other GUI grounding issues.

除了之前讨论过的GPT-4o的局限性外，还需强调其他几个GUI定位问题。

### 45 Non-standard UI and Noise.

#### 46 非标准用户界面和干扰。

Non-standard UI, be it in visual style or in behaviour, can lead GPT-4o to misinterpret UI item functionality and application context state. The same applies to visual noise in the form of update pop-up, external contents (e.g., ads), new e-mail/chat messages, etc.

无论是视觉风格还是行为上的非标准用户界面，都可能导致GPT-4o误解界面元素的功能和应用上下文状态。同样，更新弹窗、外部内容（如广告）、新邮件/聊天消息等视觉干扰也会产生类似影响。

CapCut is affected by both factors, as further illustrated in Figure 53. Moreover, its UI includes non-standard layouts involving precise positioning and drag/dropping. Lack of such prior knowledge by GPT4-o and differences in behaviour between similar functions, may also lead to mistakes in

CapCut同时受到这两方面因素的影响，如图53所示。此外，其界面包含涉及精确定位和拖放的非标准布局。GPT-4o缺乏此类先验知识，以及类似功能间行为差异，也可能导致错误。



Figure 52: Visual behaviour in Outlook that may lead GPT-4o to visual understanding mistakes.

图52：Outlook中的视觉行为，可能导致GPT-4o在视觉理解上出错。

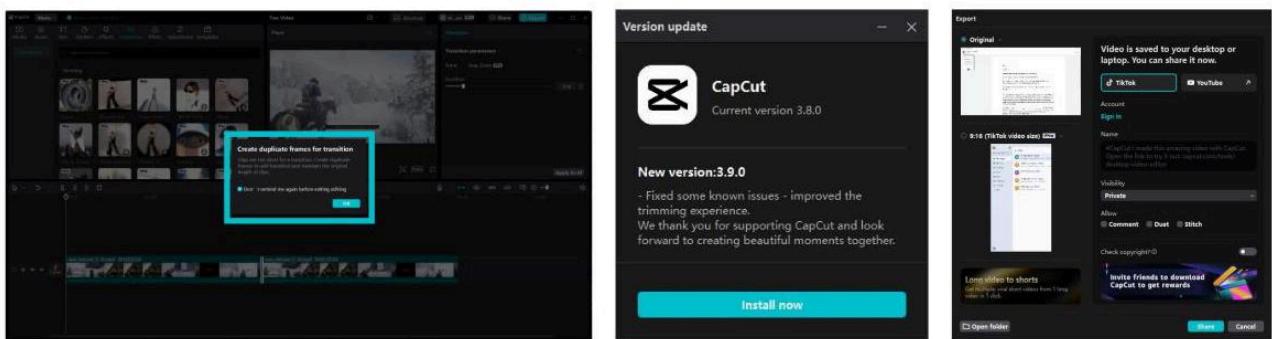


Figure 53: Different CapCut pop-ups

图53：不同的CapCut弹窗

trying to decompose actions to perform. E.g., "Add an effect" requires very different UI-interaction depending on details. Users can add effects in three different ways: i) dragging an effect to the timeline; ii) click the plus sign in a given effect in the effects panel, which adds the effect to the current place on the timeline; and iii) drag an effect directly onto a video and apply the effect to the entire video.

尝试分解要执行的操作。例如，“添加效果”根据具体情况需要非常不同的界面交互。用户可以通过三种方式添加效果：i) 将效果拖到时间线；ii) 点击效果面板中某个效果的加号，将效果添加到时间线当前位置；iii) 将效果直接拖到视频上，应用于整个视频。

## 47 Visual Context Detail.

### 48 视觉上下文细节。

GPT-4o still struggles with detailed visual understanding and over-relies on textual information or hallucinations, which results in insufficient attention to visual context and leads to understanding and reasoning mistakes.

GPT-4o在细节视觉理解方面仍存在困难，过度依赖文本信息或产生幻觉，导致对视觉上下文关注不足，从而引发理解和推理错误。

One such common example is GPT-4o declaring a dialog state to be ready to press a button like "Save", while ignoring no file name was provided, even if GPT-4o has been prompted to check for such situations. The same applies to it suggesting keyboard shortcuts to open menus that do not exist in the image being interpreted, e.g., trying to press 'Alt + F' to open the "File" menu on a screenshot that has no "File" menu.

一个常见例子是GPT-4o宣称对话框状态已准备好按下“保存”按钮，却忽略了未提供文件名，即使已提示其检查此类情况。同样，它会建议使用键盘快捷键打开图像中不存在的菜单，例如尝试按“Alt + F”打开截图中没有的“文件”菜单。

Lastly, this lack of attention to context details can also affect understanding the outcome of operations over visual content, leading to incorrect estimation of operation success, e.g., when retouching an image or deciding between a circle and a heart for a shape form.

最后，这种对上下文细节的忽视也会影响对视觉内容操作结果的理解，导致对操作成功与否的错误判断，例如修饰图像时或在圆形与心形之间选择形状时。

## 49 G OSWorld

### 50 G OSWorld

#### 50.1 G.1 Introduction to OSWorld

#### 50.2 G.1 OSWorld简介

OSWorld is a scalable, computer environment designed for multimodal agents. This platform provides a unified environment for assessing open-ended computer tasks involving various applications.

OSWorld是一个可扩展的计算机环境，专为多模态代理设计。该平台为评估涉及多种应用的开放式计算机任务提供了统一环境。

#### 50.3 G.2 OSWorld Tasks

#### 50.4 G.2 OSWorld任务

OSWorld is a benchmark suite of 369 real-world computer tasks (mostly on an Ubuntu Linux environment, but including a smaller set on Microsoft Windows) collected from authors and diverse sources such as forums, tutorials, guidelines. Each task is annotated with a natural language instruction and a manually crafted evaluation script for scoring.

OSWorld是一个包含369个真实计算机任务的基准套件（主要基于Ubuntu Linux环境，也包括少量Microsoft Windows任务），任务来源包括作者及论坛、教程、指南等多样渠道。每个任务均附有自然语言指令和手工编写的评分脚本。

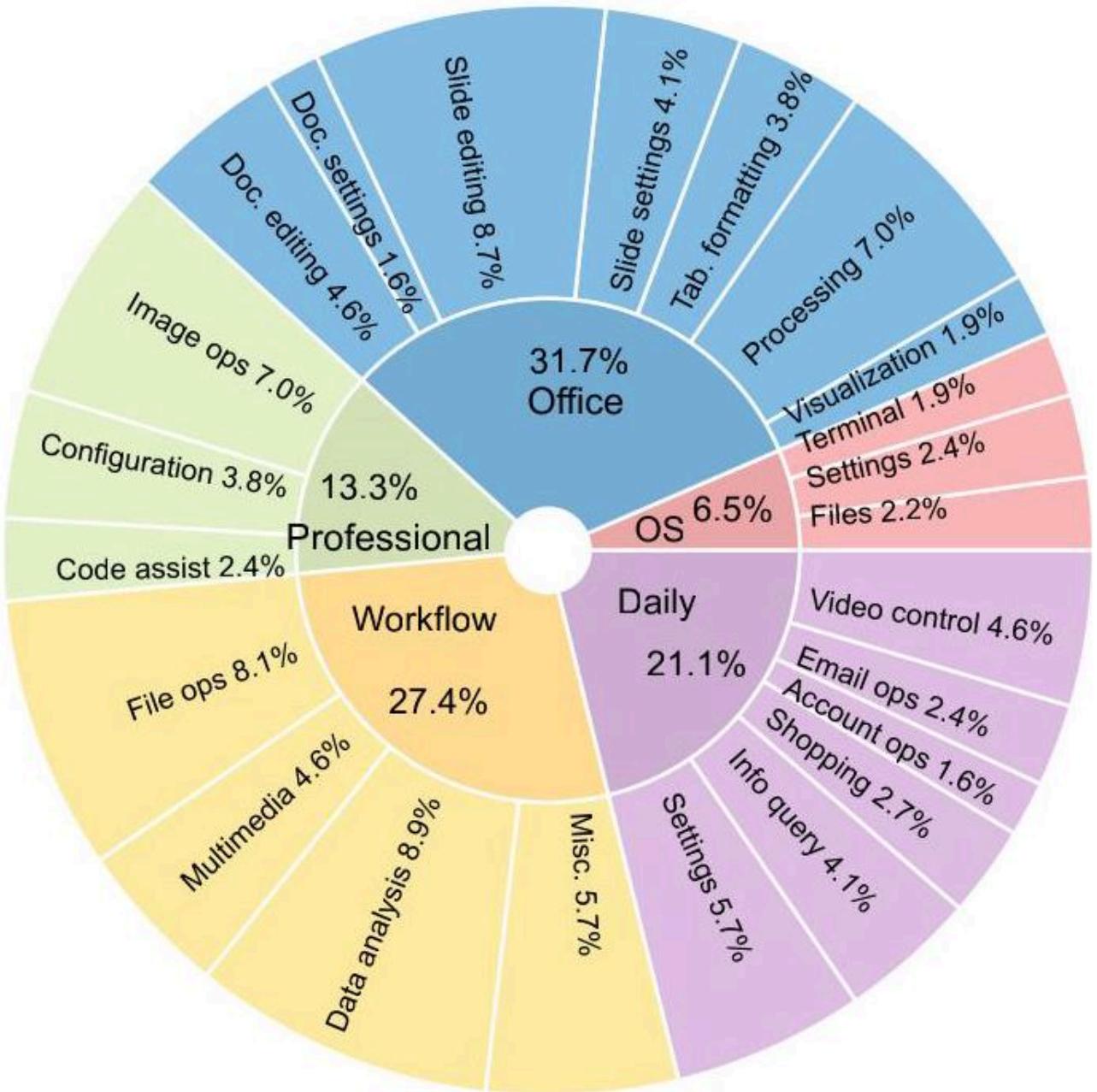


Figure 54: Task instructions distribution in OSWorld [66]

图54：OSWorld[66]中的任务指令分布

## 50.5 G.3 Implementation Details

### 50.6 G.3 实现细节

The OSWorld environment uses a virtual machine that takes in Python scripts based on PyAutoGUI for actions and provides screenshots and an accessibility tree for observations. We strictly follow the GCC settings. Our agent only uses the screenshot as input and outputs Python scripts using PyAutoGUI methods to control the keyboard and mouse (these operations are analogous to the regular action space for CRADLE). All 369 tasks use a same set of prompt templates.

OSWorld环境使用一个虚拟机，接收基于PyAutoGUI的Python脚本作为操作指令，并提供截图和无障碍树作为观察

信息。我们严格遵循GCC设置。我们的代理仅使用截图作为输入，输出使用PyAutoGUI方法控制键盘和鼠标的Python脚本（这些操作类似于CRADLE的常规动作空间）。所有369个任务使用同一套提示模板。

We employ GPT-40 as the framework's backbone model. We use the default experimental settings, as in OSWorld's baseline agent. The executable action space is the same as the OSWorld setting, the atomic skills are as follows:

我们采用GPT-40作为框架的主模型。使用与OSWorld基线代理相同的默认实验设置。可执行动作空间与OSWorld设置相同，原子技能如下：

- Mouse Actions

- 鼠标操作

- move\_mouse\_to\_position ( $x, y$ ) : Moves the mouse to a specified position on the screen.

- move\_mouse\_to\_position ( $x, y$ ) : 将鼠标移动到屏幕上的指定位置。

- click\_at\_position ( $x, y$ ) : Performs a click at a specified position.

- click\_at\_position ( $x, y$ ) : 在指定位置执行点击操作。

- mouse\_down(button): Presses the specified mouse button.

- mouse\_down(button): 按下指定的鼠标按钮。

- mouse\_up(button): Releases the specified mouse button.

- mouse\_up(button): 释放指定的鼠标按钮。

- right\_click ( $x, y$ ) : Right-clicks at the specified position.

- right\_click ( $x, y$ ) : 在指定位置执行右键点击。

- double\_click\_at\_position ( $x, y$ ) : Double-clicks at the specified position.

- double\_click\_at\_position ( $x, y$ ) : 在指定位置执行双击。

- mouse\_drag( $x, y$ ): Drags the cursor to the position.

- mouse\_drag( $x, y$ ): 将光标拖动到指定位置。

- scroll(direction, amount): Scrolls the mouse wheel up or down by a specified amount.

- scroll(direction, amount): 按指定数量向上或向下滚动鼠标滚轮。

- Keyboard Actions

- 键盘操作

- type\_text(text): Types the specified text.

- type\_text(text): 输入指定文本。

- press\_key(key): Presses and releases the specified key.

- press\_key(key): 按下并释放指定按键。

- key\_down(key): Holds a specified key.

- key\_down(key): 按住指定的键。

- key\_up(key): Releases a specified key.

- key\_up(key): 释放指定的键。

- press\_hotkey(keys): Presses a combination of keys and releases them in the opposite order (e.g., Ctrl+C), useful for shortcuts.

- press\_hotkey(keys): 按下组合键并按相反顺序释放（例如，Ctrl+C），适用于快捷键操作。

## 51 Task Status

### 52 任务状态

- `task_is_not_feasible()`: Indicates that the task cannot be completed, providing feedback for scenarios where the agent encounters infeasible tasks.
- `task_is_not_feasible()`: 表示任务无法完成，针对代理遇到不可行任务的场景提供反馈。

Many of these basic skills require GPT-4o to directly output an (x,y) position based on a screenshot. Given that the current GPT-4o is not able to achieve such precise control, we use a grounding tool to augment the screenshot. This way, GPT-4o only needs to choose an object ID. With the object ID and the bounding box of the object, we automatically convert it to the (x,y) position needed for skill execution. Instead of having GPT-4o directly choose the executable skills that require (x,y) position input, we provide several skills that only require a label ID as input for GPT-4o.

许多基础技能需要GPT-4o根据截图直接输出一个(x,y)位置。鉴于当前GPT-4o尚无法实现如此精确的控制，我们使用定位工具对截图进行增强。这样，GPT-4o只需选择一个对象ID。通过对象ID及其边界框，我们自动转换为技能执行所需的(x,y)位置。我们不让GPT-4o直接选择需要(x,y)位置输入的可执行技能，而是提供若干仅需标签ID作为输入的技能供GPT-4o使用。

- Actions with Grounding Tools
- 使用定位工具的操作
- `click_on_label(label_id)`: Clicks on a specified label in the grounding result.
- `click_on_label(label_id)`: 点击定位结果中的指定标签。
- `double_click_on_label(label_id)`: Double-clicks on a specified label in the grounding result.
- `double_click_on_label(label_id)`: 双击定位结果中的指定标签。
- `hover_over_label(label_id)`: Moves the mouse to hover over a specified label in the grounding result.
- `hover_over_label(label_id)`: 将鼠标移动至定位结果中的指定标签悬停。
- `mouse_drag_to_label(label_id)`: Drags the mouse to a specified label in the grounding result.
- `mouse_drag_to_label(label_id)`: 将鼠标拖动至定位结果中的指定标签。

Information Gathering. Tasks in OSWorld require pixel-level mouse control. While GPT-4 exhibits grounding ability, using tools like SAM can further augment the screenshot with the grounding of icons in complex computer control tasks. The bounding box is helpful for GPT-4 to understand the occurrence of objects on the screen and can also be used to calculate the precise position for mouse control.

信息收集。OSWorld中的任务需要像素级的鼠标控制。虽然GPT-4具备定位能力，但使用如SAM (Segment Anything Model) 等工具可以进一步增强截图中图标的定位，适用于复杂的计算机控制任务。边界框有助于GPT-4理解屏幕上对象的出现位置，也可用于计算鼠标控制的精确位置。

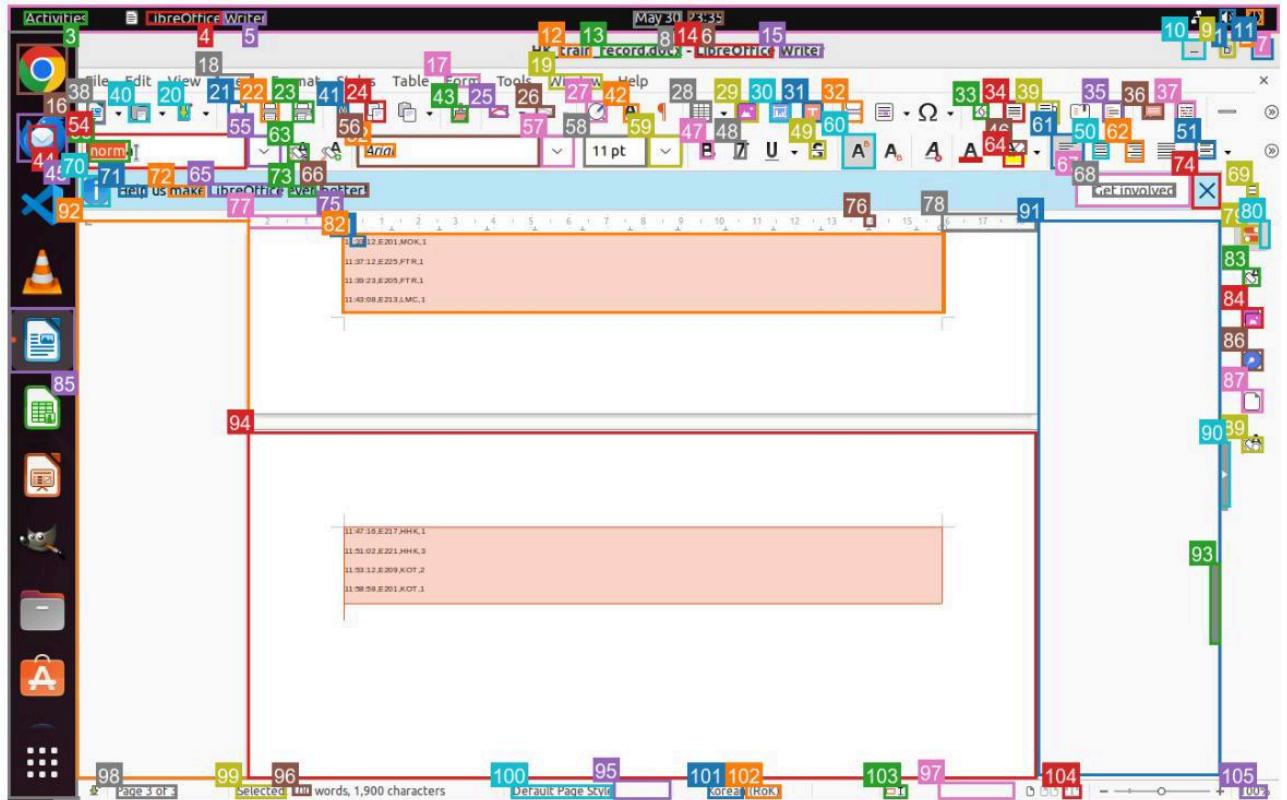


Figure 55: Augmented screenshot using CRADLE's SAM2SOM

图55：使用CRADLE的SAM2SOM增强的截图

Self-Reflection. The reflection module evaluates whether previous actions have been successfully executed and determines if the entire task was successful. The self-reflection module is important for tasks in OSWorld, which are sequential decision-making problems that require re-planning based on the current state and previous actions. The self-reflection module also helps to identify infeasible tasks.

自我反思。反思模块评估先前操作是否成功执行，并判断整个任务是否成功。反思模块对于OSWorld中的任务尤为重要，这些任务是基于当前状态和先前操作需要重新规划的序列决策问题。反思模块还帮助识别不可行的任务。

## 52.1 G.4 Application Target and Setting Challenges

### 52.2 G.4 应用目标与设置挑战

Evaluations within OSWorld reveal notable challenges in agents' abilities, particularly in GUI understanding and operational knowledge [66]. To further complete tasks in OSWorld, the agent needs advanced visual capabilities and robust GUI interaction abilities. Furthermore, the agents

OSWorld中的评估揭示了代理能力的显著挑战，尤其是在图形用户界面（GUI）理解和操作知识方面[66]。为了进一步完成OSWorld中的任务，代理需要具备高级视觉能力和强健的GUI交互能力。此外，代理

face challenges in leveraging lengthy raw observation and action records. The next-level approach encompasses designing more effective agent architectures that augment the agents' abilities to explore autonomously and synthesize their findings.

在利用冗长的原始观察和动作记录时面临挑战。下一阶段的方法包括设计更有效的智能体架构，以增强智能体自主探索和综合其发现的能力。

## 52.3 G.5 Case Studies

### 52.4 G.5 案例研究

#### 52.4.1 G.5.1 Information Gathering

#### 52.4.2 G.5.1 信息收集

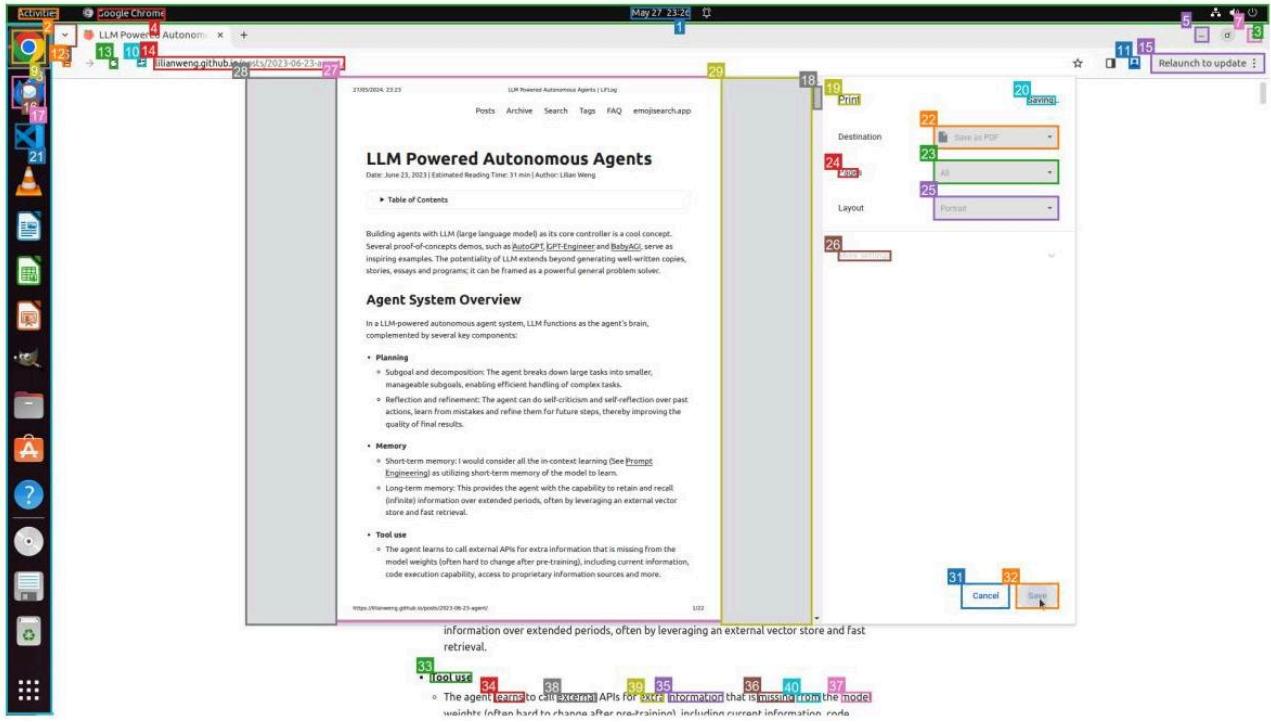


Figure 56: Case Study of robust and precise GUI interaction via information gathering

图56：通过信息收集实现稳健且精确的图形用户界面（GUI）交互的案例研究

With SAM as the grounding tool, we prompt the agent to identify the objects in each bounding box to determine the exact position of each object. As shown in Figure 56, the agent recognized the GUI element in box 32 as the Save button. In the planner, the agent chose to click on box 32 to save the PDF, resulting in success.

以SAM作为定位工具，我们提示代理识别每个边界框中的对象，以确定每个对象的准确位置。如图56所示，代理将32号框中的GUI元素识别为保存按钮。在规划器中，代理选择点击32号框以保存PDF，操作成功。

#### 52.4.3 G.5.2 Planning with Self-reflection

#### 52.4.4 G.5.2 结合自我反思的规划

We showcase how self-reflection combined with planning helps the agent complete a task by coming up with an alternative plan and validating its success.

我们展示了自我反思与规划相结合如何帮助代理通过提出替代方案并验证其成功来完成任务。

The current task instruction is "Copy the file 'file1' to each of the directories 'dir1', 'dir2', 'dir3'." As shown in Figure 57, the agent made two attempts at implementing the command but encountered errors and warnings.

当前任务指令是“将文件‘file1’复制到目录‘dir1’、‘dir2’、‘dir3’中。”如图57所示，代理尝试执行该命令两次，但遇到了错误和警告。

As shown in Figure 58, after observing the errors and warnings in the previous steps, the agent checked the files in the directory to debug. After confirming the file structure, the agent tried different commands.

如图58所示，在观察到前述步骤中的错误和警告后，代理检查了目录中的文件以进行调试。确认文件结构后，代理尝试了不同的命令。

As shown in Figure 59, after executing the new command without receiving an error message, the agent checks whether the files have been copied to the folders. After observing the result, it marks this task as a success.

如图59所示，在执行新命令且未收到错误信息后，代理检查文件是否已复制到各文件夹。观察结果后，代理将该任务标记为成功。

## 52.5 G.6 Quantitative Evaluation

### 52.6 G.6 定量评估

The detailed success rates for each application are listed in Table 16. We followed the same experimental settings as the OSWorld paper, running the experiment only once. Our results show that our agent performs better in the Chrome and GIMP domains. However, the difference in performance in the OS, Writer, and VSC domains is less statistically significant due to the smaller number of tasks. While improved information gathering and self-reflection empowered the agent in these domains,

各应用的详细成功率列于表16。我们遵循与OSWorld论文相同的实验设置，实验仅运行一次。结果显示，我们的代理在Chrome和GIMP领域表现更佳。然而，由于任务数量较少，代理在操作系统（OS）、Writer和VS Code领域的性能差异统计学意义较小。尽管改进的信息收集和自我反思增强了代理在这些领域的能力，

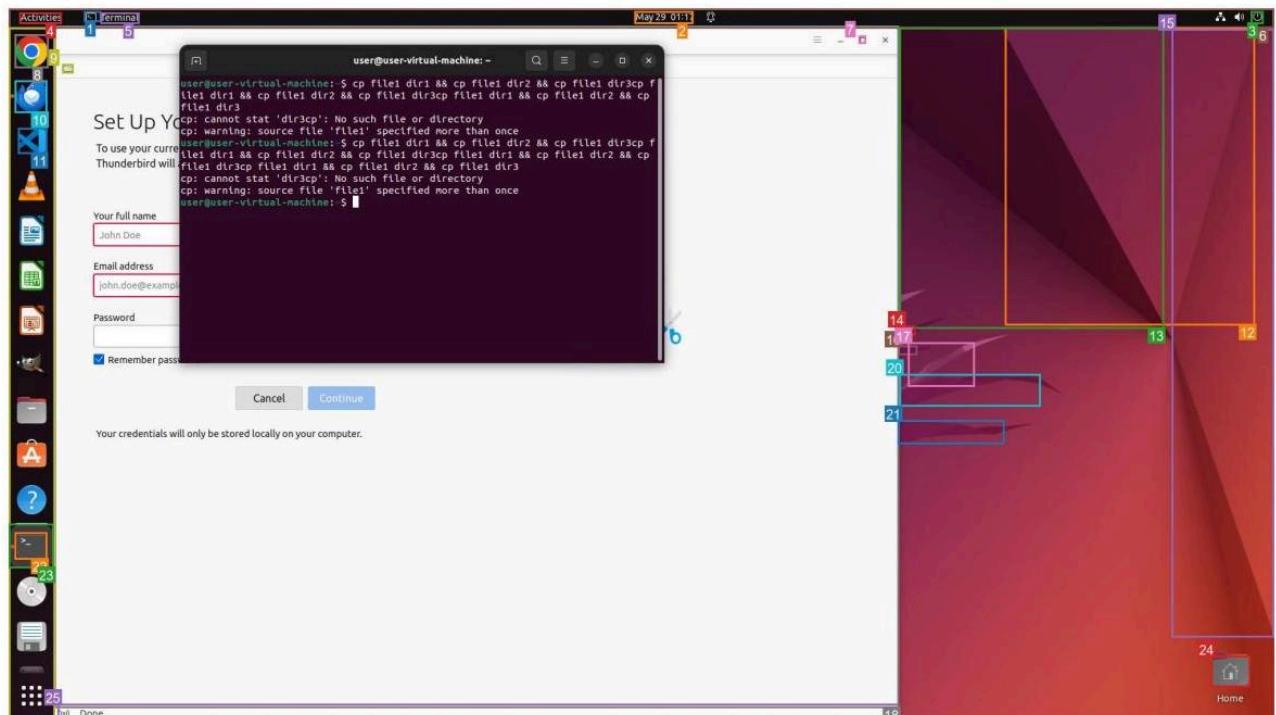


Figure 57: The agent fails to copy the files due to using incorrect commands

图57：代理因使用错误命令而未能复制文件

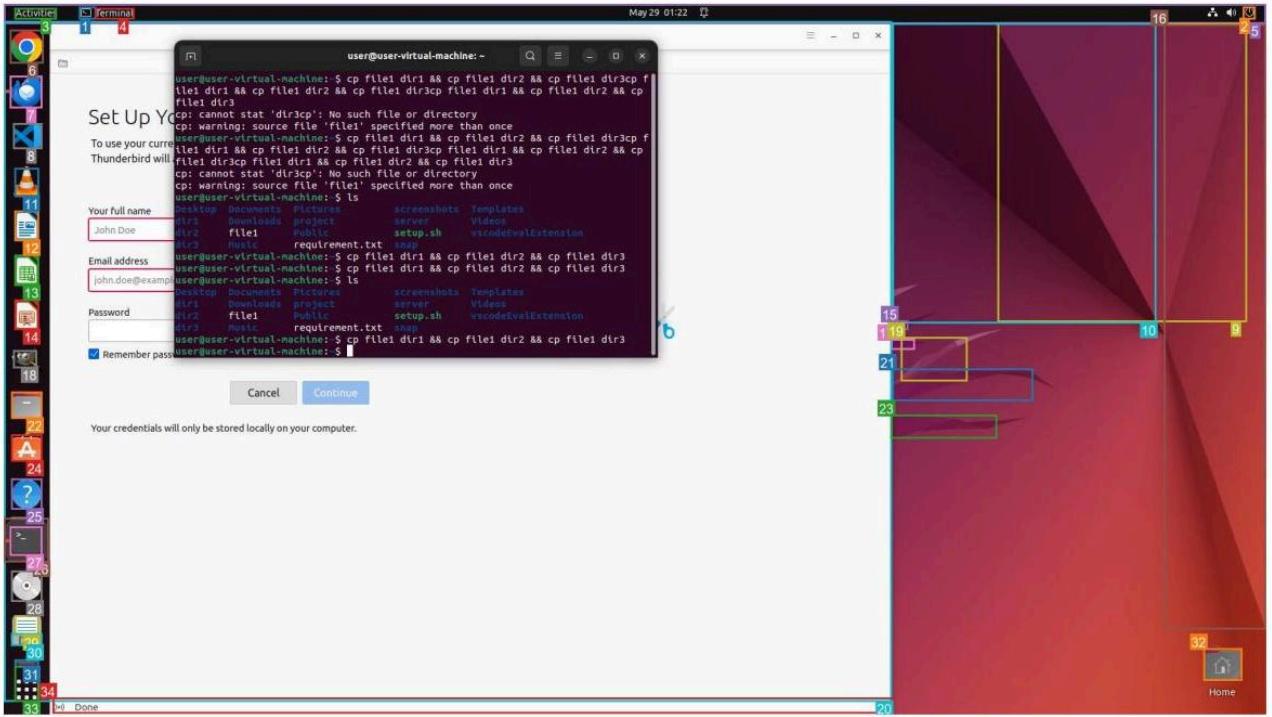


Figure 58: The agent reflects on the errors, checks the file structure and tries to debug

图58：代理反思错误，检查文件结构并尝试调试

Table 16: Detailed success rates divided by domains: OS, LibreOffice Calc, LibreOffice Impress, LibreOffice Writer, Chrome, VLC Player, Thunderbird, VS Code, GIMP, and Workflow (i.e., involves multiple applications).

表16：按领域划分的详细成功率：操作系统（OS）、LibreOffice Calc、LibreOffice Impress、LibreOffice Writer、Chrome、VLC播放器、Thunderbird、VS Code、GIMP及工作流（即涉及多个应用）。

Method	OS (24)	Calc (47)	Impress (47)	Writer (23)	VLC (17)	TB (15)	Chrome (46)	VSC (23)	GIMP (26)	Workflow (101)
GPT-40	8.33	0.00	6.77	4.35	16.10	0.00	4.35	4.35	3.85	5.58
GPT-40+SoM	20.83	0.00	6.77	4.35	6.53	0.00	4.35	4.35	0.00	3.60
Cradle	16.67	0.00	4.65	8.70	6.53	0.00	8.70	0.00	38.46	5.48

方法	操作系 统 (OS) (24)	电子表 格 (Calc) (47)	演示文稿 (Impress) (47)	文字处理 (Writer) (23)	VLC 播放器 (17)	TB (15)	Chrome 浏览器 (46)	Visual Studio Code (VSC) (23)	GIMP 图 像编辑器 (26)	工作流程 (Workflow) (101)
GPT-40	8.33	0.00	6.77	4.35	16.10	0.00	4.35	4.35	3.85	5.58
GPT-40+SoM	20.83	0.00	6.77	4.35	6.53	0.00	4.35	4.35	0.00	3.60
Cradle	16.67	0.00	4.65	8.70	6.53	0.00	8.70	0.00	38.46	5.48

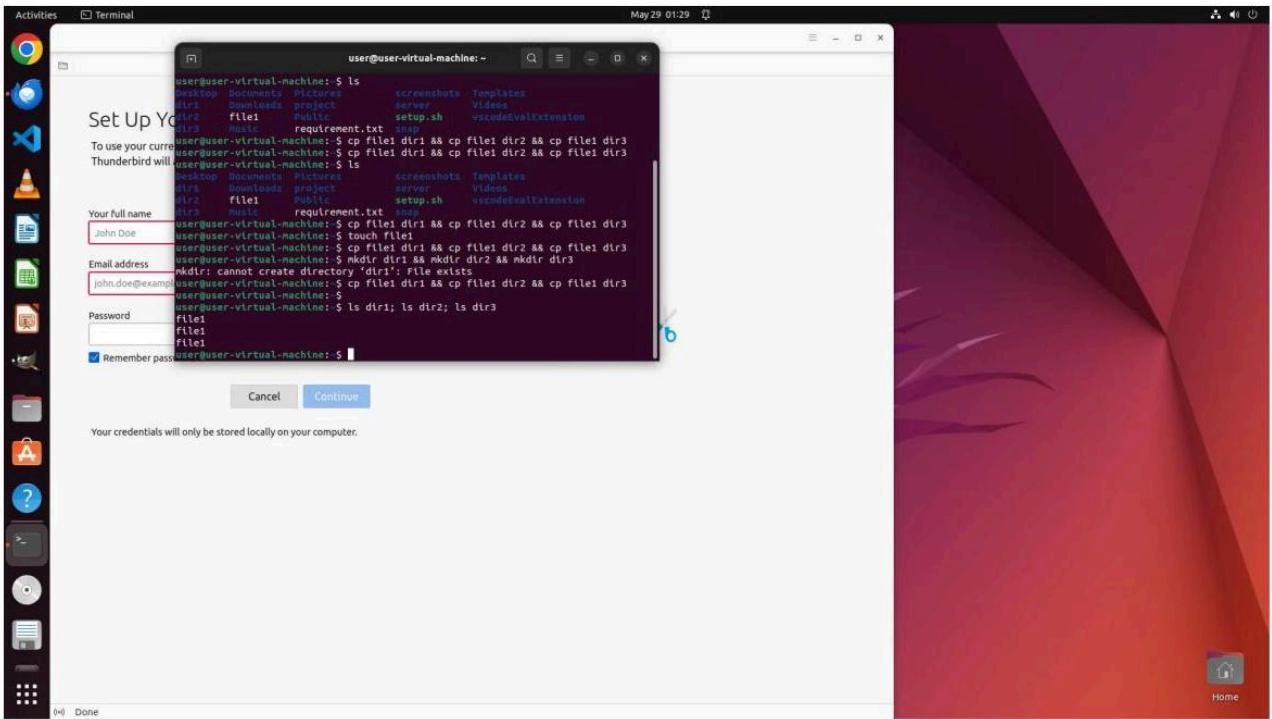


Figure 59: The agent checks if the files have already been copied

图59：代理检查文件是否已被复制

the complex pipeline and limitations of current grounding tools and GPT-4 hindered performance in domains like VLC and VSC. We identify these limitations as future directions for implementing the agent in real-world scenarios. 复杂的流程和当前定位工具及GPT-4的局限性阻碍了在VLC和VSC等领域的性能表现。我们将这些局限性确定为未来在实际场景中实现该代理的方向。

## 53 H Cradle Prompts

### 54 H Cradle 提示

Here we exemplify the utilized prompts, for each module in the framework. All prompts and customizations are included in the relevant branch in CRADLE's open-source repository in GitHub<sup>12</sup>.

这里展示了框架中各模块所用的示例提示。所有提示和自定义内容均包含在CRADLE开源仓库的相关分支中，GitHub<sup>12</sup>。

#### 54.1 H.1 Prompts for RDR2

##### 54.2 H.1 RDR2提示

Prompt 1: RDR2: Information Gathering prompt.

提示1：RDR2：信息收集提示。

Assume you are a helpful AI assistant integrated with 'Red Dead

假设你是一名集成于PC版《荒野大镖客救赎2》（Red Dead Redemption 2）中的智能助手，

1 | Redemption 2' on the PC, equipped to handle a wide range of tasks  
2 | 能够处理游戏中的各种任务。

1 in the game. Your advanced capabilities enable you to process and  
2 你具备先进的能力，能够处理和  
1 interpret gameplay screenshots and other relevant information.  
2 解析游戏截图及其他相关信息。

<\${few\_shots}>

<\${image\_introduction}>

Current task:

当前任务：

<\${task\_description}>

Target\_object\_name: Assume you can use an object detection model to

目标对象名称：假设你可以使用目标检测模型

1 detect the most relevant object for completing the current task if  
2 检测完成当前任务所需的最相关对象。  
1 needed. What object should be detected to complete the task based  
2 基于当前截图和任务，应检测哪个对象以完成任务?  
1 on the current screenshot and the current task? You should obey  
2 你应遵守以下规则：  
1 the following rules:

1. The object should be relevant to the current target or the
2. 该对象应与当前目标或当前任务的中间目标相关。

1 intermediate target of the current task. Just give one name  
2 只需给出一个名称，不带任何修饰词。  
1 without any modifiers.

2. If no explicit weapon is specified on the weapon radial menu,
3. 如果武器轮盘菜单中未明确指定武器，

1 prioritize choosing 'gun' as the weapon.  
2 优先选择"枪"作为武器。

---

---

<sup>12</sup> <https://github.com/BAAI-Agents/Cradle>

<sup>12</sup> <https://github.com/BAAI-Agents/Cradle>

---

3. If no explicit shoot target is specified, prioritize choosing '

4. 如果未明确指定射击目标，优先选择“人”作为目标。

1 | person' as the target.

4. If no explicit item is specified, only output 'null'.

5. 如果未明确指定物品，仅输出“null”。

5. If the object name belongs to the person type, replace it with '

6. 如果对象名称属于人物类型，替换为“人”。

1 | person'.

6. If there is no need to detect an object, only output "null".

7. 如果不需要检测对象，仅输出“null”。

7. If you are on the trade, map, inventory, or satchel interfaces,

8. 如果处于交易、地图、物品栏或背包界面，

1 | only output 'null',  
2 | 仅输出"null"。

Reasoning\_of\_object: Why was this object chosen, or why is there no

对象推理：为何选择该对象，或为何无需检测对象？

1 | need to detect an object?

Description: Please describe the screenshot image in detail. Pay

描述：请详细描述截图中的图像。注意

1 | attention to any maps in the image, if any, especially critical  
2 | 图像中的任何地图，尤其是关键

1 | icons, red paths to follow, or created waypoints. If there are  
2 | 图标、红色路径或已创建的路径点。如果有

1 | multiple images, please focus on the last one.  
2 | 多张图片，请重点关注最后一张。

Screen\_classification: Please select the class that best describes the

屏幕分类：请选择最能描述该截图的类别，选项包括“物品栏”、“环形菜单”、“背包”、“地图”、

1 screenshot among "Inventory", "Radial menu", "Satchel", "Map", "  
2 "交易"、"暂停"和"无菜单的通用游戏界面"。

1 Trade", "Pause", and "General game interface without any menu".  
2 请在Screen\_classification输出中给出截图的类别。

1 Output the class of the screenshot in the output of

1 Screen\_classification.

Reasoning\_of\_screen: Why was this class chosen for the current  
屏幕分类理由：为什么为当前截图选择了该类别？

1 screenshot?

Movement: Does the current task require the character to go somewhere?

移动：当前任务是否需要角色前往某处？

Noun\_and\_Verb: The number of nouns and verbs in the current task.  
名词和动词：当前任务中的名词和动词数量。

Task\_horizon: Please judge the horizon of the current task, i.e.,

任务时长：请判断当前任务的时长，即

1 whether this task needs multiple or only one interaction.  
2 该任务是否需要多次交互或仅一次交互。

There are two horizon types: long-horizon and short-horizon. For long-  
任务时长分为两种类型：长时长和短时长。对于长时长

1 horizon tasks, the output should be 1. For short-horizon tasks,

2 对于长远任务，输出应为1。对于短期任务，

1 the output should be 0 . You should obey the following rules:

2 输出应为0。您应遵守以下规则：

1. If the task contains only nouns without verbs, it is short-hori:

2. 如果任务仅包含名词而无动词，则为短期任务：

2. If the task contains more than one verb, it is long-horizon.

3. 如果任务包含多个动词，则属于长远任务。

3. If the task requires the character to go somewhere, it is long-

4. 如果任务要求角色前往某地，则时间较长-

1 horizon.  
2 地平线。

Short-horizon tasks are sub-goals during a long-horizon task, which  
短期任务是在长期任务中的子目标，

1 only need one interaction. There are some examples of short-  
2 只需一次交互。有一些简短的示例-

1 horizon tasks:  
2 地平线任务：

1. Pick up something: To complete this task, the character needs to
2. 拾取物品：要完成此任务，角色需要

1 execute the action "pick up" only once, so it is short-horizon.  
2 只执行一次"拾取"动作，因此是短期的。

2. Use or press [B] key: The character needs to press the key [B] only
3. 使用或按下[B]键：角色只需按下[B]键

1 once to talk, so it is short-horizon.  
2 只谈一次，所以视野较短。

3. Talk to somebody: The character needs to press a certain button
4. 与某人交谈：角色需要按下特定按钮

1 once to complete this task, so it is short-horizon.  
2 只需一次即可完成此任务，因此是短期的。

Long-horizon tasks are long-term goals, which usually need many  
长期任务是需要较长时间完成的目标，通常需要许多

1 interactions. There are some examples of long-horizon tasks.  
2 交互。有一些长远任务的例子。

1. Go outside: The character should go outside step by step, so it is
2. 走到外面：角色应一步一步走到外面，因此这是

1 long-horizon.  
2 长远的。

2. Approach something: The character should move closer to the target
3. 接近某物：角色应一步步靠近目标，

1 step by step, so it is long-horizon.  
2 因此这是长远的。

3. Keep away from something, shoot, take down, or battle with
4. 远离某物、射击、击倒或与某物战斗：

1 something: The character must engage in a series of interactions,  
2 角色必须进行一系列交互，

1 so it is long-horizon.  
2 因此这是长远的。

Reasoning\_of\_task: Why do you make such a judgment of task\_horizon?

任务推理：你为何做出这样的任务远景判断？

You should only respond in the format described below and not output  
你应仅以以下描述的格式回复，不要输出

1 comments or other information.  
2 评论或其他信息。

Target\_object\_name :

目标对象名称:

Name

名称

Reasoning\_of\_object:

对象推理:

1. ...

2. ...

1 | ...

Description:

描述:

The image shows...

图像显示...

Screen\_classification:

屏幕分类:

1 | Class of the screenshot  
2 | 截图类别

Reasoning\_of\_screen:

屏幕推理:

1 | 1. ...

2. ...

1 | ...

Movement:

移动:

1 Yes or No

2 是或否

1 Noun\_and\_Verb:

2 名词和动词:

1 1 noun 1 verb

2 1个名词 1个动词

1 Task\_horizon:

2 任务范围:

1 1

1 Reasoning\_of\_task:

2 任务推理:

1 1. ...

2. ...

1 | ...

---

Prompt 2: RDR2: Gather Text Information prompt.

提示2: RDR2: 收集文本信息提示。

---

Assume you are a helpful AI assistant integrated with 'Red Dead

假设你是一个集成在PC版《荒野大镖客救赎2》(Red Dead Redemption 2)中的智能助手,

1 'Redemption 2' on the PC, equipped to handle a wide range of tasks  
2 能够处理游戏中的各种任务。

1 in the game. Your advanced capabilities enable you to process and  
2 你具备先进的能力, 可以处理和

1 interpret gameplay screenshots and other relevant information.  
2 解析游戏截图及其他相关信息。

<\$image\_introduction\$>

Information: List all text prompts on the screenshot from the top to

信息: 从上到下列出截图中的所有文本提示, 即使文本提示只有一个词。

```
1 | the bottom, even the text prompt is one word.
```

All information should be categorized into one or more kinds of <  
所有信息应被分类为一种或多种<

```
1 | \$information_type\$. If you think a piece of information is both "  
2 | \$information_type\$. 如果你认为一条信息同时属于"  
  
1 | A" and "B" categories, you should write information in both "A"  
2 | "A"和"B"类别，则应将信息写入"A"  
  
1 | and "B" categories. For example, "use E to drink water" could both  
2 | 和"B"类别。例如，"用E喝水"既可以是  
  
1 | be "Action Guidance" and "Task Guidance" categories.  
2 | "行动指导"也可以是"任务指导"类别。
```

Item\_status: The helpful information to the current context in the

物品状态：对当前游戏环境有用的信息，如现金、弹药数量、当前使用的物品、玩家是否通缉等。

```
1 | game, such as the cash, amount of ammo, current using item, if the  
2 | 这些内容应为状态名称及其对应数值的配对。例如，"现金: 100\$"。如果屏幕上没有文本且无物品状态，仅输出"null"。  
  
1 | player is wanted, etc. This content should be pairs of status  
  
1 | names and their values. For example, "cash: 100\$". If there is no  
  
1 | on-screen text and no item status, only output "null".
```

Environment\_information: The information about the location, time,

环境信息：关于位置、时间、天气等的信息。这些内容应为状态名称及其对应数值的配对。例如，“位置：  
VALENTINE”。如果屏幕上没有文本且无环境信息，仅输出“null”。

```
1 | weather, etc. This content should be pairs of status names and  
  
1 | their values. For example, "location: VALENTINE". If there is no  
  
1 | on-screen text and environment information, only output "null".
```

Notification: The game will give notifications showing the events in

通知：游戏会发出通知，显示世界中的事件，如获得物品或奖励、完成

```
1 | the world, such as obtaining items or rewards, completing
```

1 objectives, and becoming wanted. Besides, it also contains  
2 <objectives, and becoming wanted. Besides, it also contains>目标，并变得受欢迎。此外，它还包含  
</objectives, and becoming wanted. Besides, it also contains>

1 valuable notifications of the game's mechanisms, such as "Health  
2 游戏机制的重要提示，例如"生命值

1 is displayed in the lower left corner". The content must be the on  
2 显示在左下角"。内容必须是开

1 -screen text. If there is no on-screen text or notification, only  
2 -屏幕文字。如果没有屏幕文字或通知，则仅

1 output "null".  
2 输出"null"。

Task\_guidance: The content should obey the following rules:

任务指导：内容应遵守以下规则：

1. The content of task guidance must be an on-screen text prompt,
2. 任务指导内容必须为屏幕上的文本提示，

1 including the menu and the general game interface.  
2 包括菜单和整体游戏界面。

2. The game will give guidance on what should be done to proceed with
3. 游戏将指导应采取的步骤以继续进行

1 the game, for example, "follow Tom". This is task guidance.  
2 例如，游戏中的"跟随汤姆"。这就是任务指导。

---

---

3. The game will give guidance on how to perform a task using keyboard
4. 游戏将指导如何使用键盘执行任务

1 keys or mouse buttons, for example, "use E to drink water". This  
2 按键或鼠标按钮，例如，"按E键喝水"。这

1 is task guidance.  
2 是任务指导。

4. If no on-screen text prompt or task guidance exists, only output "
5. 如果没有屏幕上的文本提示或任务指导，则仅输出

1 null". Never derive the task guidance from the dialogue or  
2 "null"。切勿从对话中推导任务指导或

1 notifications.  
2 通知。

Action\_guidance: The game will give guidance on how to perform a task

操作指导：游戏将提供如何执行任务的指导

1 using keyboard keys or mouse buttons; you must generate the code  
2 使用键盘按键或鼠标按钮；您必须生成代码

1 based on the on-screen text. The content of the code should obey  
2 基于屏幕上的文本。代码内容应遵守

1 the following code rules:

2 以下代码规则：

1. You should first identify the exact keyboard or mouse key

2. 您应首先确定具体的键盘或鼠标按键

1 represented by the icon on the screenshot. 'Ent' refers to 'enter'  
2 由截图中的图标表示。“Ent”指的是“enter”（回车键）

1 '. 'RM' refers to 'right mouse button'. 'LM' refers to 'left mouse'  
2 "RM"指"右键鼠标", "LM"指"左键鼠标"

1 button'. You should output the full name of the key in the code.  
2 按钮。你应该在代码中输出该键的完整名称。

2. You should refer to different examples strictly based on the word

3. 你应严格根据单词参考不同的例子

1 used to control the key, such as 'use', 'hold', 'release', 'press'  
2 用于控制按键，例如“使用”、“按住”、“释放”、“按下”

1 ', and 'click'.  
2 ', 和 '点击'.

3. If 'use' or 'press' is in the prompt to control the keyboard key or

4. 如果提示中包含“use”或“press”来控制键盘按键或

1 mouse button, io\_env.key\_press ('key', 2) or io\_env.mouse\_click ('  
2 鼠标按钮, io\_env.key\_press ('key', 2) 或 io\_env.mouse\_click ('

1 button', 2) must be used to act on it. Refer to Examples 1, 2, and  
2 必须使用<button', 2>来操作它。请参阅示例1、2和

1 3.

4. If there are multiple keys, io\_env.key\_press ('key1,key2', 2) must

5. 如果有多个按键, io\_env.key\_press ('key1,key2', 2) 必须

1 be used to act on it. Refer to Example 4.  
2 用于对此采取行动。参见示例4。

5. If 'hold' is in the prompt to control the keyboard key or mouse

6. 如果提示中包含“hold”以控制键盘按键或鼠标

1 button, it means keeping the key held with io\_env.key\_hold or the  
2 按钮，这意味着使用 io\_env.key\_hold 保持按键按下状态，或者

1 button held with io\_env.mouse\_hold (usually indefinitely, with no  
2 按钮由 io\_env.mouse\_hold 按住（通常是无限期按住，无

1 duration). If you need to hold it briefly, specify a duration  
2 持续时间）。如果需要短暂按住，请指定持续时间

1 argument. Refer to Examples 5 and 6.  
2 论证。参见示例5和6。

6. All durations are set to a minimum of 2 seconds by default. You can

7. 所有持续时间默认设置为最少2秒。您可以

1 choose a longer or shorter duration. If it should be indefinite,  
2 选择较长或较短的持续时间。如果应为无限期，

1 do not specify a duration argument.  
2 不要指定持续时间参数。

7. The name of the created function should only use phrasal verbs,

8. 创建的函数名称应仅使用短语动词，

1 verbs, nouns, or adverbs shown in the prompt and should be in the  
2 提示中显示的动词、名词或副词应当出现在

1 verb+noun or verb+adverb format, such as drink\_water,  
2 动词+名词或动词+副词格式，例如 drink\_water (喝水) ,

1 slow\_down\_car, and ride\_faster. Note that words that do not show  
2 减慢车速，加快骑行速度。注意那些未显示的词

1 in the prompt are prohibited.  
2 <prompt>中禁止使用。</prompt>

This is Example 1. If "press" is in the prompt and the text prompt on

这是示例1。如果提示中包含“press”，且文本提示为

1 the screenshot is "press \$X\$ to play the card", your output should  
2 截图显示"按\$X\$出牌"，你的输出应为

1 be :

Coopython

Coopython

def play\_card():

def play\_card():

1    ""

1 press "x" to play the card  
2 按"x"出牌

1    ""

1 io\_env.key\_press('x', 2)  
2 io\_env.key\_press('x', 2)

ccc

ccc

This is Example 2. If the instructions involve the mouse and the text

这是示例2。如果说明涉及鼠标且截图中的文本提示是

1 | prompt on the screenshot is "use the left mouse button to confirm  
2 | "使用左键确认", 你的输出应为:

1 | ", your output should be:

<<opthon

<<opthon

def confirm():

def confirm():

1 | III II

1 | use "left mouse button" to confirm  
2 | 使用"左键"确认

1 | III II

1 | io\_env.mouse\_click("left mouse button")  
2 | io\_env.mouse\_click("left mouse button")

ccc

ccc

This is Example 3. If "use" is in the prompt and the text prompt on

这是示例3。如果提示中包含“use”，且截图中的文本提示是

1 | the screenshot is "use ENTER to drink water", your output should  
2 | "使用回车键喝水", 你的输出应为

1 | be :

ccpython

ccpython

def drink\_water():

def drink\_water():

1 | \*\*\*\*

1 | use "enter" to drink water  
2 | 按"回车"喝水

1 | \*\*\*\*

```
1 | io_env.key_press('enter', 2)
2 | io_env.key_press('enter', 2)
```

ccc  
ccc

This is Example 4. If "use" is in the prompt and the text prompt on

这是示例4。如果提示中包含“use”，且截图中的文本提示是“use W and J to jump the barrier”，则你的输出应为：

```
1 | the screenshot is "use W and J to jump the barrier", your output
2 | 截图中的文本提示是"使用W和J跳过障碍"，你的输出应为：
```

```
1 | should be:
2 | 应为：
```

44 python  
44 python

```
def jump_barrier():
def jump_barrier():
```

```
1 | III II
```

```
1 | use "w" and "j" to jump the barrier
2 | 使用"w"和"j"跳过障碍
```

```
1 | """
```

```
1 | io_env.key_press('w, j', 3)
2 | io_env.key_press('w, j', 3)
```

ccc  
ccc

This is Example 5. If "hold" is in the prompt and the text prompt on

这是示例5。如果提示中包含“hold”，且截图中的文本提示是“hold H to run”，则你的输出应为：

```
1 | the screenshot is "hold H to run", your output should be:
2 | 截图中的文本提示是"按住H键跑"，你的输出应为：
```

```
<<opthon
<<opthon
```

```
def run():
def run():
```

```
1 | III II
```

```
1 | hold "h" to run
2 | 按住"h"键跑
```

```
1 | III II
```

```
1 | io_env.key_hold('h')
2 | io_env.key_hold('h')
```

ccc  
ccc

This is Example 6. If the instructions involve the mouse and the text

这是示例6。如果指令涉及鼠标且截图中的文本提示是“按住右键以聚焦目标”，你的输出应为：

```
1 | prompt on the screenshot is "hold the right mouse button to focus
```

```
1 | on the target", your output should be:
```

<<opthon  
<<opthon

```
def focus_on_target():
def focus_on_target():
```

```
1 |     """"
```

```
1 | hold "right mouse button" to focus
2 | 按住"右键"以聚焦
```

```
1 |     """"
```

```
1 | io_env.mouse_hold("right mouse button")
2 | io_env.mouse_hold("right mouse button")
```

ccc  
ccc

This is Example 7. If "release" is in the prompt and the text prompt

这是示例7。如果提示中包含“release”，且截图中的文本提示是“释放Q键以丢弃物品”，你的输出应为：

```
1 | on the screenshot is "release Q to drop the items", your output
```

```
1 | should be:
```

cccpthon  
cccpthon

```
def drop_items():
def drop_items():
```

```
1 |     """"
```

1 | release "q" to drop the items  
2 | 释放"q"键以丢弃物品  
  
1 | """

1 | io\_env.key\_release('q')  
2 | io\_env.key\_release('q')

ccc  
ccc

Dialogue: Conversations between characters in the game. This content  
对话: 游戏中角色之间的对话内容。

1 | should be in the format of "character name: dialogue". For example  
2 | 格式应为"角色名: 对话"。例如  
  
1 | , "Arthur: I'm fine". If there is no on-screen text or dialogue,  
2 | , "Arthur: 我很好"。如果没有屏幕文本或对话,  
  
1 | only output "null".  
2 | 则只输出"null"。

Other: Other information that does not belong to the above categories.  
其他: 不属于上述类别的其他信息。

1 | If there is no on-screen text, only output "null".  
2 | 如果没有屏幕文本, 则只输出"null"。

Reasoning: The reasons for classification for each piece of  
推理: 对每条信息分类的理由。

1 | information.

If the on-screen text prompt is an instruction on how to perform a  
如果屏幕文本提示是关于如何使用键盘按键或鼠标按钮执行任务的指令,

1 | task using keyboard keys or mouse buttons, it should also  
2 | 也应归类为动作指导和任务指导。  
  
1 | classified as action guidance and task guidance.

For action guidance, which code rules should you follow based on the  
对于动作指导, 应根据用于控制按键或按钮的词汇, 如按下、按住、

1 | word used to control the key or button, such as press, hold,  
2 | 释放和点击, 遵循哪些代码规则?  
  
1 | release, and click?

The information should be in the following categories, and you should  
信息应包含以下类别, 且您应

1 | output the following content without adding any other explanation:  
2 | 输出以下内容，不添加任何其他说明：

Information:

信息：

1. ...

2. ...

...

Reasoning :

推理：

1. ...

2. ...

1 | ...

1 | Item\_status:  
2 | 项目状态：

Item\_status is ...

项目状态是.....

Environment\_information:

环境信息：

Environment information is ...

环境信息是.....

Notification:

通知：

Notification is ...

通知是.....

Task\_guidance:

任务指导:

1 Task is ...  
2 任务是.....

Action\_guidance:

行动指导:

1 cccpython  
2 cccpython  
  
1 Python code to execute  
2 执行的Python代码  
  
1 ccc  
2 ccc  
  
1 44 CPython  
2 44 CPython  
  
1 Python code to execute  
2 要执行的Python代码  
  
1 ccc  
2 ccc  
  
1 ...

Dialogue:

对话:

Dialogue is ...

对话是.....

1 Other:  
2 其他:

Other information is ...

其他信息是.....

---

Prompt 3: RDR2: Self-Reflection prompt.

提示3: RDR2: 自我反思提示。

---

Assume you are a helpful AI assistant integrated with 'Red Dead

假设你是集成在PC版《荒野大镖客救赎2》(Red Dead Redemption 2)中的智能助手，

1 'Redemption 2' on the PC, equipped to handle a wide range of tasks  
2 能够处理游戏中的各种任务。

1 in the game. Your advanced capabilities enable you to process and  
2 你具备先进的能力，可以处理和

1 interpret gameplay screenshots and other relevant information.  
2 解析游戏截图及其他相关信息。

1 Your task is to examine these inputs, interpret the in-game  
2 你的任务是审查这些输入，解读游戏内

1 context, and determine whether the executed action takes effect.  
2 情境，并判断所执行的动作是否生效。

Current task:

当前任务：

<\$task\_description\$>

Last executed action:

上次执行的动作：

<\$previous\_action\$>

Implementation of the last executed action:

上一次执行动作的实现：

<\$action\_code\$>

Error report for the last executed action:

上一次执行动作的错误报告：

<\$executing\_action\_error\$>

Reasoning for the last action:

上一次动作的推理：

<\$previous\_reasoning\$>

Valid action set in Python format to select the next action:

用于选择下一动作的Python格式有效动作集：

<\$skill\_library\$>

<\$image\_introduction\$>

Reasoning: You need to answer the following questions step by step to  
推理：你需要逐步回答以下问题以

1 | get some reasoning based on the last action and sequential frames  
2 | 基于上一次动作和连续帧获取一些推理

1 | of the character during the execution of the last action.  
2 | 关于角色在执行上一次动作期间的表现。

1. What is the last executed action not based on the sequential frames
2. 根据连续帧，上一执行动作是什么？

1 | ?

2. Was the last executed action successful? Give reasons. You should
3. 上一次执行动作是否成功？请给出理由。你应当

1 | refer to the following rules:  
2 | 参考以下规则：

- If the action involves moving forward, it is considered unsuccessful
- 如果动作涉及向前移动，只有当角色位置在连续帧中保持不变时，

1 | only when the character's position remains unchanged across  
2 | 该动作才被视为不成功，无论背景元素和其他

---

1 | sequential frames, regardless of background elements and other  
2 | 人物如何。

1 | people.

3. If the last action is not executed successfully, what is the most
4. 如果上一次动作未成功执行，最可能的原因是什么？你应只给出一个原因并参考

1 | probable cause? You should give only one cause and refer to the

1 | following rules:  
2 | 以下规则：

- The reasoning for the last action could be wrong.
- 上一次操作的推理可能是错误的。
- Not holding enough time should not be considered in this part.
- 在此部分不应考虑持有时间不足的问题。

- If it is an interaction action, the most probable cause was that the
- 如果是交互动作，最可能的原因是

1 | action was unavailable or not activated at the current place.  
2 | 当前地点无法使用该操作或未激活。

- If it is a movement action, the most probable cause was that you
- 如果是移动动作，最可能的原因是你

1 | were blocked by seen or unseen obstacles.  
2 | 被可见或不可见的障碍物阻挡。

- If there is an error report, analyze the cause based on the report.
- 如果有错误报告，请根据报告分析原因。

You should only respond in the format as described below:

您应仅按照以下描述的格式回复：

Reasoning :

推理：

1. ...

2. ...

3. ...

...

---

Prompt 4: RDR2: Task Inference prompt.

提示4：RDR2：任务推断提示。

---

Assume you are a helpful AI assistant integrated with 'Red Dead  
假设你是一个集成在《荒野大镖客》（Red Dead）中的智能助手

1 | 'Redemption 2' on the PC, equipped to handle a wide range of tasks  
2 | 《荒野大镖客救赎2》在PC上，能够应对各种任务

1 | in the game. You will be sequentially given <\\\\$event\_count\\\$>  
2 | 在游戏中。你将依次获得<\\\\$event\_count\\\$>

1 | screenshots and corresponding descriptions of recent events. You  
2 | 最近事件的截图及相应描述。您

1 will also be given a summary of the history that happened before  
2 还将提供之前发生的历史概要

1 the last screenshot. You should assist in summarizing the events  
2 最后一张截图。你应该协助总结事件

1 for future decision-making.  
2 用于未来的决策制定。

The following are <sevent\_count\$> successive screenshots and

以下是连续的<sevent\_count\$>张截图及

1 corresponding descriptions:  
2 对应描述：

<\$image\_introduction\$>

The following is the summary of history that happened before the last

以下是上一次之前发生的历史概要

1 screenshot:  
2 截图：

<\$previous\_summarization\$>

Current task:

当前任务：

<\$task\_description\$>

Info\_summary: Based on the above input, please make a summary from the

信息摘要：基于以上输入，请做一个总结

1 screenshots with descriptions and the history in no less than 10  
2 带有描述的截图及不少于10条的历史记录

1 sentences, following the rules below.  
2 句子，遵循以下规则。

1. Summarize the tasks from the history and the current task, with a
2. 总结历史任务和当前任务，包含一个

1 special note on the method of crucial press operations.  
2 关于关键压制操作方法的特别说明。

2. Summarize the entities and behaviors mentioned in the successive
3. 总结连续提及的实体和行为

1 descriptions.  
2 描述。

3. If entities and behaviors in the history and screenshots are missed
4. 如果历史记录和截图中的实体及行为被遗漏

1 | in the descriptions, please add them to the summarization.  
2 | 在描述中, 请将它们添加到总结中。

4. Organize the summarization as a story in order of time, including

5. 按时间顺序将总结组织成一个故事, 包括

1 | the past entities and behaviors.  
2 | 过去的实体和行为。

5. Only give descriptions; do not provide suggestions.

6. 仅给出描述; 不提供建议。

Entities\_and\_behaviors: Entities and behaviors which are summarized, e

Entities\_and\_behaviors: 被总结的实体和行为, 例如

1 | .g., The entities include the player's character, the target  
2 | 实体包括玩家角色、目标角色, 以及玩家和目标的马匹。

1 | character, and horses for both the player and the target. The  
2 | 行为包括玩家角色骑马、骑马跟随目标,

1 | behaviors consist of the player character riding horseback,  
2 | 并向前移动以保持在目标身后的距离。

1 | following the target on horseback, and moving forward to maintain

1 | a distance behind the target.

The output should be in the following format:

输出格式应如下:

Info\_summary :

信息总结 :

The summary is...

总结是...

---

Prompt 5: RDR2: Action Planning prompt.

提示5: RDR2: 行动规划提示。

---

You are a helpful AI assistant integrated with 'Red Dead Redemption 2'

你是集成于PC版《荒野大镖客2》(Red Dead Redemption 2)中的智能助手,

1 | on the PC, equipped to handle various tasks in the game. Your  
2 | 具备处理游戏中各种任务的能力。

1 advanced capabilities enable you to process and interpret gameplay  
2 你的高级功能使你能够处理和解读游戏玩法信息。

1 screenshots and other relevant information. By analyzing these  
2 截图及其他相关信息。通过分析这些

1 inputs, you gain a comprehensive understanding of the current  
2 输入，您可以全面了解游戏中的当前

1 context and situation within the game. Utilizing this insight, you  
2 背景和情境。利用这些洞察，您

1 are tasked with identifying the most suitable in-game action to  
2 的任务是确定在当前任务下最合适的游戏内行动，

1 take next, given the current task. You control the game character  
2 您控制游戏角色，

1 and can execute actions from the available action set. Upon  
2 并可以从可用的动作集中执行操作。评估所提供的信息后，

1 evaluating the provided information, your role is to articulate  
2 您的角色是明确表达

1 the precise action you would deploy, considering the game's  
2 在游戏当前情况下您将采取的具体行动，

1 present circumstances, and specify any necessary parameters for  
2 并指定执行该行动所需的任何参数。

1 implementing that action.

Here is some helpful information to help you make the decision.

以下是帮助您做出决策的一些有用信息。

Current task:

当前任务：

<\$task\_description\$>

Memory examples:

记忆示例：

<\$memory\_introduction\$>

<\$few\_shots\$>

<\$image\_introduction\$>

Last executed action:

上一次执行的动作：

<\$previous\_action\$>

Reasoning for the last action:

上一次动作的推理:

<\$previous\_reasoning\$>

Self-reflection for the last executed action:

对上一次执行动作的自我反思:

<\$previous\_self\_reflection\_reasoning\$>

Summarization of recent history:

近期历史总结:

<\$info\_summary\$>

Valid action set in Python format to select the next action:

用于选择下一步动作的Python格式有效动作集:

<\$skill\_library\$>

Minimap information:

小地图信息:

<\$minimap\_information\$>

Based on the above information, you should first analyze the current

基于以上信息，你应首先分析当前

1 | situation and provide the reasoning for what you should do for the  
2 | 状况，并说明你为完成任务下一步应采取的理由。

1 | next step to complete the task. Then, you should output the exact  
2 | 然后，你应输出你想在游戏中执行的具体

1 | action you want to execute in the game. You should respond to me  
2 | 动作。你应回复我

1 | with:  
2 | 内容为：

Reasoning: You should think step by step and provide detailed

推理: 你应逐步思考并提供详细的

1 | reasoning to determine the next action executed on the current  
2 | 推理，以确定在当前任务状态下执行的下一步动作。

1 state of the task. You need to answer the following questions step  
2 你需要逐步回答以下问题。

1 by step. You cannot miss the question number 13:  
2 你不能遗漏第13号问题：

1 1. Only answer this question when the radial menu, trade, map,  
2 1. 仅当径向菜单、交易、地图、

1 satchel or inventory interfaces are open. You should first  
2 背包或库存界面打开时回答此问题。你应先

1 describe each item in the screen line by line, from the top left  
2 逐行描述屏幕上的每个物品，从左上角开始，向右移动。

1 and moving right. Is the target item in the current screen?  
2 当前屏幕中是否有目标物品？

---

---

1 2. Only answer this question when the radial menu, trade, map,  
2 2. 仅当径向菜单、交易、地图、

1 satchel or inventory interfaces are open. Which item is selected  
2 背包或物品栏界面打开时回答此问题。当前选中了哪个物品？

1 currently?  
2 当前选中了哪个物品？

1 3. Only answer this question when the character is visible in the  
2 3. 仅当角色在当前步骤的截图中可见时回答此问题。角色在

1 screenshot of the current step. Where is the character in the  
2 当前步骤的截图中处于何处？

1 screenshot of the current step?  
2 当前步骤的截图中角色处于何处？

1 4. Where is the target in the screenshot of the current step based  
2 4. 根据任务描述，目标在当前步骤的截图中是在左侧还是右侧？

1 on the task description, on the left side or on the right side?  
2 目标在当前步骤的截图中是在左侧还是右侧？

1 Does it appear in the previous screenshots?  
2 它是否出现在之前的截图中？

1 5. Are there any bounding boxes with coordinates values and object  
2 5. 截图中是否有带坐标值和物体标签的边界框，

1 labels,such as "door \$x = {0.5},y = {0.5}\$\$ ",shown in the screenshot?  
2 例如"door \$x = {0.5},y = {0.5}\$\$"?

1 The answer must be based only on the screenshot of the current  
2 答案必须仅基于当前步骤的截图，而非任何之前的步骤。如果答案是否定的，忽略

1 step, not on any previous steps. If the answer is no, ignore the  
2 第6至8题。

1 questions 6 to 8 .  
2 第6至8题。

1 6. You should first describe each bounding box, from left to right  
2 6. 你应先从左到右描述每个边界框。哪个边界框与目标更相关?

1 . Which bounding box is more relevant to the target?  
2 哪个边界框与目标更相关?

1 7. What is the value \$x\$ of the most relevant bounding box only in  
2 当前截图中最相关的边界框的值\$x\$是什么?

1 the current screenshot? The value is the central coordination (x,y  
2 该值是该框中心点的坐标 (x,y

1 ) of the central point of the box.  
2 )。

1 8. Based on the few shots and the value \$x\$ ,where is the relevant  
2 8. 基于少量样本和该值\$x\$，当前截图中相关的边界框位于哪里?

1 bounding box in the current screenshot? Clearly on the left side,  
2 明显在左侧，略偏左，中间，略偏右，还是明显在右侧?

1 slightly on the left side, in the center, slightly on the right

1 side, or clearly on the right side?

1 9. Only answer this question when the radial menu, trade, map,

2 9. 仅当径向菜单、交易、地图、背包或物品栏界面未打开时回答此问题。总结近期历史内容，主要关注历史任务和行为。

1 satchel or inventory interfaces are not open. Summarize the

1 contents of recent history, mainly focusing on the historical

1 tasks and behaviors.

1 10. Only answer this question when the radial menu, trade, map,

2 10. 仅当径向菜单、交易、地图、背包或物品栏界面未打开时回答此问题。总结最近执行动作的自我反思内容，不要被其他信息分散注意力。

1 satchel or inventory interfaces are not open. Summarize the

1 content of self-reflection for the last executed action, and do

1 | not be distracted by other information.

1 | 11. What was the previous action? If the previous action was a  
2 | 11. 上一个动作是什么？如果上一个动作是

1 | turn, was it a left or a right turn? If the previous action was a  
2 | 转弯，是左转还是右转？如果前一个动作是

1 | movement, were you blocked?

2 | 移动，你被阻挡了吗？

1 | 12. List conditions in action rule 12 and which condition is

2 | 12. 列出动作规则12中的条件以及满足了哪个条件。

1 | satisfied. Only when you do not satisfy any conditions, summarize

2 | 只有当你不满足任何条件时，才总结

1 | the content of the minimap information.

2 | 小地图信息的内容。

1 | 13. This is the most critical question. Based on the action rules

2 | 13. 这是最关键的问题。基于动作规则

1 | and self-reflection, what should be the most suitable action in  
2 | 和自我反思，下一步在有效动作集合中最合适的动作是什么？

1 | the valid action set for the next step? You should analyze the

2 | 你应逐步分析该动作的效果。

1 | effects of the action step by step.

Actions: The best action, or short sequence of actions without gaps,

动作：最佳动作，或无间断的短动作序列，

1 | to execute next to progress in achieving the goal. Pay attention  
2 | 用于下一步执行以推进目标的实现。注意

1 | to the names of the available skills and to the previous skills  
2 | 可用技能的名称以及之前已执行的技能（如有）。你还应更多关注

1 | already executed, if any. You should also pay more attention to  
2 | 以下动作规则：

1 | the following action rules:

1 | 1. You should output actions in Python code format and specify any  
2 | 1. 你应以Python代码格式输出动作，并指定执行该动作所需的任何参数。如果函数有

1 | necessary parameters to execute that action. If the function has

1 | parameters, you should also include their names and decide their  
2 | 参数时，你还应包括它们的名称并确定它们的

1 values, like "move(duration=1)". If it does not have a parameter,  
2 值, 例如"move(duration=1)"。如果没有参数,

1 just output the action, like "mount\_horse()".  
2 只需输出动作, 如"mount\_horse()"。

1 2. Given the current situation and task, you should only choose  
2 2. 根据当前情况和任务, 你应仅从有效动作集中选择

1 the most suitable action from the valid action set. You cannot use  
2 最合适的动作。不能使用不在有效动作集中的

1 actions that are not in the valid action set to control the  
2 动作来控制角色。

1 character.  
2 角色。

1 3. If the target is not on the radial menu, trade, satchel or  
2 3. 如果目标不在径向菜单、交易、背包或

1 inventory interfaces, you MUST choose the skill 'view\_next\_page'.  
2 物品栏界面, 必须选择技能"view\_next\_page"。

1 For the map, ignore the skill 'view\_next\_page'.  
2 对于地图, 忽略技能"view\_next\_page"。

1 4. If the minimap information exists, it may include angle  
2 4. 如果存在小地图信息, 可能包含红点、黄点或黄色区域的角度

1 information for red points, yellow points, or yellow regions.  
2 信息。

1 Angle information specifies the direction of the corresponding  
2 角度信息指定对应点或区域的方向。负角度表示左侧,

1 point or area. A negative angle indicates the left side, while a  
2 正值表示右侧。如果角度为30,

1 positive value signifies the right side. If the angle is 30, the  
2 对应点或区域位于角色右侧30度处。

1 corresponding point or area is 30 degrees to the character's right  
2 . If the angle is -50, the corresponding point or area is 50  
2 如果角度为-50, 则对应的点或区域为50

1 degrees to the character's left. Do not doubt the correctness of  
2 向角色的左侧旋转度数。请相信其正确性

1 these angles; you can refer to them when you approach these points  
2 这些角度; 当你接近这些点时可以参考它们

1 or regions.  
2 或地区。

1 5. When you decide to control the character to move, if the  
2 5. 当你决定控制角色移动时, 如果

1 relevant bounding box is clearly on the left side in the current  
2 相关的边界框明显位于当前的左侧

1 screenshot, you MUST turn left with a big degree. If the relevant  
2 截图时, 您必须大幅度向左转。如果相关

1 bounding box is slightly on the left side in the current  
2 当前边界框略偏左侧

1 screenshot, you MUST turn left with a small degree. If the  
2 截图时, 您必须稍微向左转。

1 relevant bounding box is clearly on the right side in the current  
2 相关的边界框明显位于当前的右侧

1 screenshot, you MUST turn right with a big degree. If the relevant  
2 截图时, 您必须大幅度向右转。如果相关

1 bounding box is slightly on the right side in the current  
2 当前边界框略偏右侧

1 screenshot, you MUST turn right with a small degree. If the  
2 截图时, 您必须稍微向右转。

1 relevant bounding box is on the central side of the current  
2 相关的边界框位于当前的中心侧

1 screenshot, you can choose to move forward.  
2 截图后, 您可以选择继续前进。

1 6. When you decide to control the character to move, if yellow  
2 6. 当你决定控制角色移动时, 如果是黄色

1 regions or yellow points exist in minimap information, they are  
2 小地图信息中存在区域或黄色点, 它们是

1 related to the current task or instruction. This implies that you  
2 与当前任务或指令相关。这意味着你

1 should approach within the yellow region or approach the yellow  
2 应接近黄色区域内或接近黄色

1 points. You can refer to the corresponding angle information when  
2 点。您可以参考相应的角度信息, 当

1 deciding to approach these regions or points. If red points exist  
2 决定接近这些区域或点。如果存在红点

1 in the minimap information, they are also related to the current  
2 在小地图信息中, 它们也与当前相关

1 task or instruction. This implies that you should turn towards  
2 任务或指令。这意味着你应该转向

1 them, and you can also refer to the corresponding angle  
2 他们, 你也可以参考相应的角度

1 information.  
2 信息。

1 7. When you decide to control the character to move, if minimap  
2 7. 当你决定控制角色移动时, 如果小地图

1 information does not exist, the 'theta' you use to turn MUST be  
2 信息不存在，你用来转动的"theta"必须是

1 more than 10 degrees and less than 60 degrees.  
2 大于10度且小于60度。

1 8. When you decide to control the character to move, if you are in  
2 8. 当你决定控制角色移动时，如果你处于

1 a normal road condition, the 'duration' you use to move forward  
2 在正常路况下，你用来前进的"持续时间"

1 should be 1 second. If you have bad road conditions, such as snow,  
2 应为1秒。如果路况不好，比如有积雪，

1 and grass, that can slow you down, the 'duration' you use to move  
2 和草地，会减慢你的速度，你用来移动的"持续时间"

1 forward should be 2 seconds.  
2 前进时间应为2秒。

1 9. When you are exploring or searching a place, if you are leaving  
2 9. 当你在探索或搜索一个地方时，如果你要离开

1 the place, you MUST make a sharp turn to face the inside of the  
2 到达该处时，您必须急转弯面向内部

1 place. Any values for degrees are allowed.  
2 地点。角度的任何数值均可。

1 10. If upon self-reflection you think the last action was  
2 10. 如果经过自我反思你认为上一个动作是

1 unavailable at the current place, you MUST move to another place.  
2 当前位置不可用，您必须移动到其他地方。

1 11. If upon self-reflection you think you were blocked, you MUST  
2 11. 如果经过自我反思你认为自己被阻止了，你必须

1 make a moderate turn in the same direction as the previous turn  
2 在与上一次转弯相同的方向上适度转弯

1 action and move forward, so that you can pass obstacles.  
2 采取行动并向前推进，这样你才能克服障碍。

1 12. The conditions to ignore the minimap information for decision-  
2 12. 忽略小地图信息以进行决策的条件

1 making are: 1. When self-reflection implies you were blocked. 2.  
2 决策时机有：1. 当自我反思表明你遇到了阻碍。2.

1 When you were inside the highlighted area in the minimap. If any  
2 当你处于小地图中高亮区域内时。如果有的话

1 of the conditions satisfied, you must ignore the minimap  
2 满足条件时，您必须忽略小地图

1 information for decision-making even if it is relevant to the  
2 决策所需的信息，即使它与...相关

1 current task.  
2 当前任务。

- 1   | 13. When you are indoors, or the current task does not imply
  - 2   | 13. 当你在室内，或当前任务不涉及
- 
- 1   | following, you MUST not use the follow action.
  - 2   | 跟随时，您不得使用跟随动作。
- 
- 1   | 14. When you are outdoors, and the current task implies following,
  - 2   | 14. 当您在户外，且当前任务涉及跟随时，
- 
- 1   | you MUST use the follow action.
  - 2   | 您必须使用跟随动作。
- 
- 1   | 15. If you were dead or the game failed, you MUST retry from the
  - 2   | 15. 如果您死亡或游戏失败，您必须从检查点重试，
- 
- 1   | checkpoint, and MUST NOT restart the mission.
  - 2   | 且不得重新开始任务。

You should only respond in the format described below, and you should

您应仅以以下描述的格式回应，且不应输出评论或其他信息：

- 1   | not output comments or other information:

Reasoning:

推理：

1. ...

2. ...

3. ...

Actions:

动作：

cccpython

cccpython

- 1   | action(args1=x,args2=y)
- 2   | action(args1=x,args2=y)

ccc

ccc

---

### 54.3 H.2 Prompts for Cities: Skylines

#### 54.4 H.2 《城市：天际线》（Cities: Skylines）提示

Prompt 6: Skylines: Information Gathering prompt.

提示6：《城市：天际线》信息收集提示。

---

Assume you are a helpful AI assistant integrated with 'Cities:

假设您是一名集成于PC版《城市：天际线》（Cities: Skylines）中的智能助手，

- 1 | Skylines on the PC, equipped to handle a wide range of tasks in
- 2 | 能够处理各种任务，
  
- 1 | the game. Your advanced capabilities enable you to process and
- 2 | 游戏。您先进的能力使您能够处理和
  
- 1 | interpret gameplay screenshots and other relevant information.
- 2 | 解读游戏截图及其他相关信息。

<\$image\_introduction\$>

Current task:

当前任务：

<\$task\_description\$>

Description: Please analyze and describe the screenshot image in

描述：请详细分析并描述截图图像，

- 1 | detail and then provide an overall image description. Pay
- 2 | 然后提供整体图像描述。请
  
- 1 | attention to anything related to the task. If there are specific
- 2 | 关注与任务相关的任何内容。如果有特定
  
- 1 | features such as characters or text, mention these as well.
- 2 | 特征，如角色或文本，也请提及。

Budget: Bank Balance is shown at the bottom of the screenshot.

预算：截图底部显示银行余额。

Population: The population of the city is shown at the bottom of the

人口：城市人口显示在截图底部，

- 1 | screenshot, next to the budget.
- 2 | 位于预算旁边。

Error\_message: If there are some in-game error messages, which are

错误信息：如果有游戏内错误信息，通常为红色，

1 | usually in red color, such as "Space already occupied!", extract  
2 | 如"空间已被占用! ", 请提取文本,

1 | the text, otherwise, only output "null".  
2 | 否则仅输出"null"。

Construction\_information: If there is some in-game construction

建筑信息: 如果有游戏内建筑信息, 通常为蓝色,

1 | information, which is usually in blue colors, such as "  
2 | 如"

1 | Construction cost: 2500 Estimated production: 0 m^3/week" and "  
2 | 建筑成本: 2500 预计产量: 0 立方米/周"及"

1 | Construction cost: 2500 Shoreline recommended", extract the text,  
2 | 建设成本: 2500 Shoreline 推荐

1 | otherwise, only output "null".  
2 | 否则, 仅输出"null"。

Other: Other information that does not belong to the above categories.

其他: 不属于上述类别的其他信息。

1 | If none of them applies, only output "null".  
2 | 如果都不适用, 仅输出"null"。

You should only respond in the format described below and not output

您应仅以以下描述的格式回复, 不要输出

1 | comments or other information.  
2 | 评论或其他信息。

Description:

描述:

The image shows...

图像显示...

Budget:

预算:

The amount of budget

预算金额

Population:

人口:

The amount of population

人口数量

Error\_message:

错误信息:

The text of the error message

错误信息文本

Construction\_information:

建设信息:

The text of the construction information

建设信息文本

Other:

其他:

Other information is

其他信息是

---

Prompt 7: Skylines: Self-Reflection prompt.

提示7: 天际线 (Skylines) : 自我反思提示。

---

Assume you are a helpful AI assistant integrated with 'Cities:

假设你是集成在PC版《城市：天际线》(Cities: Skylines)中的一个有用的AI助手，

1 | Skylines' on the PC, equipped to handle a wide range of tasks in  
2 | 能够处理游戏中的各种任务。

1 | the game. Your advanced capabilities enable you to process and  
2 | 你具备先进的能力，可以处理和

1 | interpret gameplay screenshots and other relevant information.  
2 | 解读游戏截图及其他相关信息。

1 | Your task is to examine these inputs, interpret the in-game  
2 | 你的任务是检查这些输入，解读游戏内

1 | context, and determine whether the executed action takes effect.  
2 | 的上下文，并判断所执行的操作是否生效。

Target task:

目标任务:

<\$task\_description\$>

Current subtask for completing the target task:

完成目标任务的当前子任务:

<\$subtask\_description\$>

Current coordinates:

当前坐标:

<\$coordinates\$>

Last executed action for completing the subtask:

完成子任务的最后执行操作:

<\$actions\$>

Error message for the last executed action:

最后执行操作的错误信息:

<\$error\_message\$>

Construction information:

建设信息:

<\$construction\_information\$>

Summarization of recent history:

近期历史总结:

<\$history\_summary\$>

<\$image\_introduction\$>

Reasoning: You MUST answer the following questions step by step to get

推理: 您必须逐步回答以下问题以获得

1 some reasoning based on the last action and sequential frames  
2 基于上一次动作和连续帧的一些推理

1 during the execution of the last action.  
2 在执行最后一个操作期间。

1. What is the executed action? Please answer this question not based  
2. 执行的操作是什么? 请不要基于此问题作答

1 on the sequential frames.  
2 在连续帧上。

2. Is the construction information provided in the information shown  
3. 提供的信息中是否包含施工信息

1 above? If yes, what is it?  
2 上面? 如果是, 是什么?

3. Was the last executed action successful? Give reasons. You should  
4. 上一次执行的操作成功了吗? 请说明理由。你应该

1 refer to the following rules:  
2 请参照以下规则：

- Buildings and roads cannot be built on the river.
- 河流上不能建造建筑物和道路。
- Water pumping station and water drain pipe need to be built as close
- 水泵站和排水管需要尽可能靠近建造

1 as possible to the river.  
2 尽可能靠近河边。

- If you are try\_place a water pumping station and the construction
- 如果您正在尝试建设一个水泵站及其施工

1 information provided above shows that the estimated production is  
2 上述信息显示估计产量为

1  $0 \text{ m}^3/\text{week}$ , then it means that it is not close enough to the river  
2 0 立方米/周，那么这意味着它离河流不够近

1 . So you need to try\_place to place the building to another place.  
2 所以你需要尝试使用try\_place将建筑放置到另一个位置。

1 If the estimated production is not  $0 \text{ m}^3/\text{week}$ , or the  
2 如果估计的产量不是 $0 \text{ m}^3/\text{周}$ ，或者

1 construction information is not provided, regard this action as a  
2 未提供施工信息，则视此操作为成功。

1 success. You should only refer to the textual construction  
2 您应仅参考文本施工信息，而非从连续帧中提取。

1 information instead of extracting it from the sequential frames.

- If you are try\_place a water drain pipe and the construction
- 如果您尝试放置排水管且施工信息显示推荐沿岸线，则

1 information shows that shoreline is recommended. Then it means  
2 表示距离河流不够近。因此，您需要尝试将建筑放置在其他位置。

1 that it is not close enough to the river. So you need to try\_place

1 to place the building in another place.

- Roads are prohibited from crossing together and do not build roads
- 禁止道路交叉重叠，且不得在水面上建造道路。

1 on water.

4. If the last action is not executed successfully, what is the most

5. 如果最后一次操作未成功执行，最可能的原因是什么？如何改进此操作？您应仅给出一个原因，并参照以下规

则：

1 | probable cause? How to improve this action? You should give only

1 | one cause and refer to the following rules:  
2 | - 最后一次操作的推理可能有误。

• The reasoning for the last action could be wrong.

• If there is an error message for the last executed action provided

• 如果上述信息中提供了最后执行操作的错误信息，请根据报告分析原因，

1 | in the above information, analyze the cause based on the report,

1 | otherwise, you should regard that there are no error messages. You  
2 | 否则，你应当认为没有错误信息。

1 | are not allowed to guess the error message by yourself.  
2 | 你不允许自行猜测错误信息。

5. Is the subtask completed? Give your reasons. You MUST remember that

6. 子任务完成了吗？请说明理由。你必须记住

1 | action starts with "try\_place" can NEVER complete the subtask.  
2 | 以"try\_place"开头的操作永远无法完成子任务。

1 | Only "confirm\_placement()" can make the building happen and  
2 | 只有"confirm\_placement()"才能使建筑完成并

1 | complete the task. If you want to make any confirmation, regard it  
2 | 完成任务。如果你想进行任何确认，都应视为成功。

1 | as a success.  
2 | 视为成功。

6. Do you think the subtask is reasonable? Give your reasons.

7. 你认为子任务合理吗？请说明理由。

Success: You need to output whether the last action was executed

成功：你需要输出最后一个操作是否执行成功。

1 | successfully or not.  
2 | 成功与否。

• If the last action is successful, you should only output 'True'.  
• 如果最后一个操作成功，你只应输出'True'。

1 | Otherwise, you should only output 'False'.  
2 | 否则，你只应输出'False'。

You should only respond in the format described below.

你应仅以以下描述的格式回应。

Reasoning:

推理:

1. ...

2. ...

3. ....

4. ...

5. ...

6. ...

1 | ...

1 | Success:  
2 | 成功:

1 |      True  
2 |      True

1 |      ...

---

Prompt 8: Skylines: Task Inference prompt.

提示8: 天际线: 任务推断提示。

---

Assume you are a helpful AI assistant integrated with 'Cities':

假设你是一个集成在“Cities”的有用AI助手:

1 | 'Skylines' on the PC, equipped to handle a wide range of tasks in  
2 | PC上的《天际线》, 能够处理各种任务

1 | the game. You will also be given a summary of the history that  
2 | 游戏。您还将获得一份历史概要, 内容包括

1 | happened before the last screenshot. You should assist in  
2 | 发生在最后一个截图之前。你应该协助  
1 | summarizing the events for future decision-making and also propose  
2 | 总结事件以供未来决策，并提出  
1 | a new subtask, which is the most suitable subtask for the current  
2 | 一个新的子任务，这是当前最合适子任务  
1 | situation, given the target task.  
2 | 情况，鉴于目标任务。

Here is some helpful information to help you do the summarization and  
这里有一些有助于你进行总结的有用信息

1 | propose the subtask.  
2 | 提出子任务。

Current task:

当前任务：

<\$task\_description\$>

Previous proposed subtask for the task:

之前为该任务提出的子任务：

<\$subtask\_description\$>

Previous reasoning for proposing the subtask:

提出子任务的先前推理：

<\$subtask\_reasoning\$>

<\$image\_introduction\$>

Current budget:

当前预算：

<\$budget\$>

Current population:

当前人口：

<\$population\$>

Last executed action:

上次执行的操作：

<\$actions\$>

Self-reflection for the last executed action:

对上一次执行操作的自我反思:

<\$self\_reflection\_reasoning\$>

Error message for the last action:

上一次操作的错误信息:

<\$error\_message\$>

The following is the summary of history that happened before the last

以下是上一次之前发生的历史摘要

1 | screenshot:  
2 | 截图：

<\$previous\_summarization\$>

The task can be decomposed into the following subtasks:

该任务可以分解为以下子任务:

1. Start from the Highway entry: Build a road from the highway entry
2. 从高速公路入口开始: 从高速公路入口处建造一条道路

1 | in grid (4, 2) vertically northwards towards grid (3,1).  
2 | 在网格(4, 2)垂直向北通向网格(3,1)。

2. Extend Horizontally to the Left (1,1) : From the endpoint in grid
3. 向左水平延伸(1,1): 从网格

1 | \$\left( \{1,1\} \right)\$, construct a road horizontally to the left, spanning across  
2 | \$\left( \{1,1\} \right)\$的终点, 水平向左建造一条道路, 跨越

1 | grids \$\left( \{3,1\} \right)\$ and \$\left( \{2,1\} \right)\$, and ending at the center of grid  
2 | \$\left( \{1,1\} \right)\$ .  
2 | 网格\$\left( \{3,1\} \right)\$和\$\left( \{2,1\} \right)\$, 并在网格\$\left( \{1,1\} \right)\$中心结束。

3. Build a Road Down to the bottom of Grid (2,2) : Start from grid
4. 向下建造道路至网格(2,2)底部: 从网格

1 | \$\left( \{1,1\} \right)\$ and construct the road to the top of grid \$\left( \{2,3\} \right)\$ .  
2 | \$\left( \{1,1\} \right)\$开始, 建造道路至网格\$\left( \{2,3\} \right)\$顶部。

---

---

4. Extend Eastward to Grid (3,3): From the bottom of grid (2,2), build

5. 向东延伸至网格(3,3): 从网格(2,2)底部, 向东建造

1 a road eastward to reach the center of grid  $\left( \{3,3\} \right)$  .  
2 一条道路, 达到网格 $\left( \{3,3\} \right)$ 中心。

5. Connect the road to the Highway Exit: Extend the end of the road

6. 将道路连接至高速公路出口: 将道路末端

1 from grid  $\left( \{3,3\} \right)$  to the exit of the highway, completing the road  
2 从网格 $\left( \{3,3\} \right)$ 延伸至高速公路出口, 完成道路建设

1 loop.  
2 循环。

6. Install a Water Pumping Station near the River at the top-left

7. 在左上角河流附近安装水泵站

1 corner of grid  $\left( \{2,3\} \right)$  : Place the water pumping station near the  
2 网格 $\left( \{2,3\} \right)$ 左上角: 将水泵站设置在网格 $\left( \{2,3\} \right)$ 的河流附近, 确保充足的  
水源供应。

1 river in grid  $\left( \{2,3\} \right)$  to ensure an adequate water supply.  
2 网格 $\left( \{2,3\} \right)$ 的河流旁, 以保证水源充足。

7. Position a Water Drain Pipe near the River at the top-left corner

8. 在网格(2,3)左上角河流附近布置排水管

1 of grid (2,3): Install a water drain pipe slightly downstream from  
2 在水泵站下游稍远但同一网格内安装排水管, 以防止水源污染。

1 the pumping station but within the same grid to prevent water  
2 水泵站下游但同一网格内安装排水管, 防止水污染。

1 contamination.  
2 防止水污染。

8. Lay Water Pipes: Connect the water pumping station to the water

9. 铺设水管: 用水管连接水泵站和排水管。同时, 确保所有道路均铺设水管,

1 drain pipe using water pipes. Additionally, ensure all roads built  
2 以便整个区域都能获得供水。

1 are covered with water pipes to provide water access across the  
2 确保所有道路均铺设水管, 提供全区域供水。

1 entire area.  
2 整个区域的供水。

9. Erect Wind Turbines for Power: Construct several wind turbines near

10. 建造风力发电机: 在水泵站附近及道路沿线建造多台风力发电机,

1 the water pumping station and along the roads to provide  
2 为该区域提供可持续电力。

1 sustainable electricity to the area.  
2 为该区域提供可持续电力。

10. Designate Residential Zones: Allocate spaces adjacent to the roads

11. 规划住宅区：在道路旁划定住宅用地。

1 | for residential zones to foster community living.

2 | 为住宅区营造社区生活环境。

11. Establish Industrial Zones: Set aside areas near the roads for

12. 设立工业区：在靠近道路的区域划出工业用地，

1 | industrial purposes, ideally in parts of the grid further from

2 | 最好位于远离住宅区的网格部分，

1 | residential zones to manage noise and pollution.

2 | 以控制噪音和污染。

12. Create Commercial Zones: Develop commercial zones near the roads

13. 创建商业区：在道路附近发展商业区，

1 | to provide services and retail options for the residents and

2 | 为居民和区域内的工作人员提供服务和零售选择。

1 | workers in the area.

13. Make sure all the zones near roads are built with Residential

14. 确保所有靠近道路的区域均建设为住宅区、工业区或工业区。

1 | Zones, Industrial Zones or Industrial Zones.

14. Build more roads and zones and ensure water and electricity supply

15. 建设更多道路和区域，并确保供水和供电。

History\_summary: Summarize what happened in the past experience,

历史总结：总结过去的经验，

1 | especially the last step according to the decision-making

2 | 特别是根据决策推理和自我反思推理对最后执行步骤的总结。

1 | reasoning and self-reflection reasoning for the last executed

2 | 总结需精准、具体，且高度关联任务，遵循以下规则。

1 | action. The summarization needs to be precise, concrete and highly

1 | related to the task and follow the rules below.

2 | 1. 总结历史任务和当前任务。内容包括

1. Summarize the tasks from the history and the current task. What is

1 | the current progress of the task?

2 | 任务的当前进展如何？

2. Which subtask has been completed? Which subtasks are not?

3. 哪些子任务已完成？哪些子任务尚未完成？

Subtask\_reasoning: According to the task decomposition, analyze the

子任务推理：根据任务分解，逐步分析

1 current progress step by step and then decide whether the previous  
2 当前进展，然后判断前一个

1 subtask is finished and whether it is necessary to propose a new  
2 子任务是否完成，以及是否有必要提出新的

1 subtask. The subtask should be straightforward, contribute to the  
2 子任务。子任务应当简明，能促进

1 target task and be most suitable for the current situation, which  
2 目标任务的完成，并且最适合当前情况，

1 should be completed within a few actions. You should respond to me  
2 应在几步操作内完成。你应当回复我

1 with:

2 内容包括：

1. What is the previous subtask? Which step it is for in the task

2. 前一个子任务是什么？它对应任务分解中的哪一步？

1 decomposition?

2 2. 根据自我反思的推理，前一个

2. According to the reasoning of self-reflection, is the previous

子任务是否完成？注意，动作的成功并不

1 subtask completed? Note that the success of the action does not  
2 等同于子任务的成功。你应严格遵循

1 mean the success of the subtask. You should strictly follow the  
2 自我反思中对子任务是否完成的推理。如果完成，

1 reasoning of whether the subtask is completed in the self-  
2 你应进入下一步并提出

1 reflection. If yes, you should move to the next step and propose  
2 新的子任务。

1 it as the new subtask. If not, you should continue the previous  
2 将其视为新的子任务。如果不是，您应继续之前的

1 subtask without changing anything. Please do not make any  
2 子任务，且不做任何更改。请勿做出任何

1 assumptions if they are not mentioned in the above information.  
2 未在上述信息中提及的假设。

1 You should assume that you are doing the task from scratch. Please  
2 您应假设自己是从零开始执行该任务。请

1 strictly follow the description and requirements in the current  
2 严格遵循当前任务中的描述和要求。

1 task.

2 任务。

3. The proposed subtask needs to be precise and concrete within one

4. 所提出的子任务需在一句话内准确具体。

1 sentence. It should not be related to any skills.

2 不应涉及任何技能。

4. To enable water supply, you should first build a water pumping

5. 为实现供水，您应先建造一个水泵站，

1 station and then build a water drain pipe near the river, and

2 然后在河边建造排水管，

---

---

1 finally use water pipes to connect them with the roads. And ensure  
2 最后用水管将它们与道路连接，并确保

1 the water pipes cover all the roads.

2 水管覆盖所有道路。

5. The water pumping station and water drain pipe also need

6. 水泵站和排水管也需要电力才能运行，

1 electricity to work. So you also need to provide electricity for  
2 因此您还需为它们提供电力。

1 them.

6. If you want to build roads for the village at the beginning, make

7. 如果您想一开始为村庄修建道路，

1 sure to mention that the road needs to be as long as possible and  
2 务必提到道路需要尽可能长，且

1 use several roads to form a large square for the village.

2 使用多条道路形成村庄的大广场。

Subtask: According to the subtask reasoning, determine and output the

子任务：根据子任务推理，确定并输出

1 most suitable subtask for the current situation. You MUST output

2 当前情况最合适的子任务。你必须输出

1 the subtask in the output.

2 子任务内容。

You should only respond in the format described below, and you should

你应仅以以下描述的格式回复，且

1 not output comments or other information.

2 不得输出评论或其他信息。

History\_summary:

历史摘要：

The summary is ...

摘要是.....

Subtask\_reasoning :

子任务推理：

1. ...

2. ...

3. ...

Subtask:

子任务：

The current subtask is ...

当前子任务是.....

---

Prompt 9: Skylines: Action Planning prompt.

提示9：天际线（Skylines）：行动规划提示。

---

You are a helpful AI assistant integrated with 'Cities: Skylines' on

你是集成于PC版《城市：天际线》（Cities: Skylines）中的智能助手，

1 | the PC, equipped to handle various tasks in the game. Your  
2 | 具备处理游戏中各种任务的能力。你的

1 | advanced capabilities enable you to process and interpret gameplay  
2 | 高级功能使你能够处理和解读游戏玩法

1 | screenshots and other relevant information. By analyzing these  
2 | 截图及其他相关信息。通过分析这些

1 | inputs, you gain a comprehensive understanding of the current  
2 | 输入，您可以全面了解游戏中的当前

1 | context and situation within the game. Utilizing this insight, you  
2 | 背景和情境。利用这些洞察，您

1 | are tasked with identifying the most suitable in-game action to  
2 | 的任务是根据当前任务，确定最合适的下一步游戏操作。

1 take next, given the current task. You control the game character  
2 您控制游戏角色，能够执行可用动作集中的动作。

1 and can execute actions from the available action set. Upon  
2 在评估所提供的信息后，您的角色是明确

1 evaluating the provided information, your role is to articulate  
2 您将采取的具体行动，考虑游戏的

1 the precise action you would deploy, considering the game's  
2 当前状况，并指定执行该动作所需的任何参数。

1 present circumstances, and specify any necessary parameters for  
1 implementing that action.

Here is some helpful information to help you make the decision.

以下是帮助您做出决策的一些有用信息。

Current task:

当前任务：

<\$subtask\_description\$>

Coordinates of constructed buildings:

已建建筑的坐标：

<\$coordinates\$>

The latest successful action that builds the building. If you want to

最新成功的建造建筑动作。如果您想

1 try\_place a road, and the endpoint (x2, y2), of the latest  
2 尝试放置道路，且最新成功动作的终点 (x2, y2)

1 successful action is also try\_place a road. Then you MUST use the  
2 也是尝试放置道路，那么您必须使用

1 end point of the constructed road as the start point of your new  
2 将已建道路的终点作为新道路的起点。

1 road.  
2 道路。

<\$last\_success\_try\_place\_action\$>

Current budget:

当前预算：

<\$budget\$>

Current population:

当前人口：

<\$population\$>

Last executed action:

上一次执行的操作：

<\$actions\$>

Self-reflection reasoning for the last executed action:

对上一次执行操作的自我反思推理：

---

---

<\$self\_reflection\_reasoning\$>

Error message for the last action:

上一次操作的错误信息：

<\$error\_message\$>

Construction information for the last action:

上一次操作的建设信息：

<\$construction\_information\$>

Summarization of recent history:

近期历史总结：

<\$history\_summary\$>

Valid action set in Python format to select the next action:

用于选择下一步操作的Python格式有效操作集：

<\$skill\_library\$>

<\$image\_introduction\$>

Based on the above information, analyze the current situation and  
基于以上信息，分析当前情况并

1 provide the reasoning for what you should do for the next step to  
2 提供完成任务下一步应采取行动的推理。

1 complete the task. Then, you should output the exact action you  
2 然后，你应输出你想在游戏中执行的具体操作。

1 want to execute in the game. You should respond to me with:  
2 你应回复我：

Reasoning: You should think step by step and provide detailed

推理：你应逐步思考并提供详细的

1 reasoning to determine the next action executed on the current  
2 推理，以确定当前应执行的下一步操作。

1 state of the task. You need to answer the following questions step  
2 任务的状态。您需要逐步回答以下问题

1 by step. You cannot miss the last question:  
2 逐步进行。你绝不能错过最后一个问题：

1 1. What is the current task? What are the requirements to achieve  
2 1. 当前任务是什么？实现该任务的要求有哪些？

1 the goal?  
2 目标？

1 2. According to the self-reflection reasoning, is the last action  
2 2. 根据自我反思推理，最后的动作是

1 executed successfully?  
2 执行成功了吗？

1 3. If you want to place anything, do you already open the  
2 3. 如果你想放置任何东西，你已经打开了

1 corresponding menu? Otherwise, you need to open the right menu  
2 对应的菜单？否则，您需要打开正确的菜单

1 first in this step rather than doing anything else. If you have  
2 首先在这一步，而不是做其他任何事情。如果你有

1 not already opened the corresponding menu, skip answering  
2 如果尚未打开相应菜单，则跳过回答

1 questions 4, 5, 6, 7, 8 and 9.  
2 问题4、5、6、7、8和9。

1 4. Does the previous action "try\_place" something? If there is an  
2 4. 上一个动作"try\_place"放置了什么东西吗？如果有

1 error message showing that the space is already occupied or the  
2 错误信息显示该空间已被占用或

1 last action failed according to the self-reflection reasoning, you  
2 根据自我反思推理，最后的操作失败了，你

1 should use the same action with different parameters as the  
2 应使用相同的操作但参数不同，如同

1 position of it to try again. The difference needs to be  
2 重新尝试它的位置。差异需要是

1 significant enough with at least 100 pixels of change for the  
2 至少有100像素的变化，足够显著

1 position of the input points. If there is no error message, you  
2 输入点的位置。如果没有错误信息，您

1 should only output confirm\_placement() or cancel\_placement() to  
2 应仅输出 confirm\_placement() 或 cancel\_placement() 到

1 approve or cancel the placement. You should not call anything else  
2 批准或取消该职位。你不应调用其他任何操作

1 5. Does the previous action open any menu? Then you should "  
2 5. 上一个操作是否打开了任何菜单？那么你应该"

1 try\_place" something according to the task description instead of  
2 根据任务描述"尝试放置"某物，而不是

1 using "confirm\_placement".  
2 使用"confirm\_placement"。

1 6. If you want to place a building, which grid do you plan to  
2 6. 如果你想建造一座建筑，你打算放在哪个格子上

1 place the building in? What is the exact pixel position of it?  
2 将建筑物放置在哪里？它的确切像素位置是多少？

1 7. If you want to place a road, which grids do you plan to make it  
2 7. 如果你想修建一条道路，你打算在哪些格子上建造？

1 cross? Which grids are the start point and end point in,  
2 交叉？起点和终点分别在哪些网格中，

1 respectively? What are the exact pixel positions of them? You MUST  
2 分别是？它们的确切像素位置是多少？你必须

1 use one of the endpoints of the constructed road shown in the  
2 使用所示构建道路的一个端点

1 coordinates information as the start point of the new road. If you  
2 坐标信息作为新道路的起点。如果您

1 want to try\_place a road, and the endpoint (x2, y2), of the  
2 想要尝试铺设一条道路，终点为 (x2, y2)，

1 latest successful action is also try\_place a road. Then you MUST  
2 最新成功的操作也是尝试放置一条道路。然后你必须

1 use the end point of the constructed road as the start point of  
2 使用已建道路的终点作为

1 your new road.  
2 新道路的起点。

1 8. If you want to place a zone, which grids do you plan to make it  
2 8. 如果你想设置一个区域，计划覆盖哪些网格?  
  
1 cover? You should only use the vertices coordinates of the  
2 你应仅使用对应网格的顶点坐标作为操作参数。  
  
1 corresponding grids as the parameter for the action. Zones cannot  
2 区域之间不能相互覆盖。  
  
1 cover each other.  
2 9. 如果你想铺设水管，起点应为

1 9. If you want to place a Water Pipe, the start point should be  
2 水泵站 (Water Pumping Station)、排水管 (Water Drain Pipe)、已建水管的起点或终点位置。

1 the position of Water Pumping Station, Water Drain Pipe, the start

1 point of a built Water Pipe or the end point of a built Water  
2 Pipe.

1 10. This is the most critical question. Based on the action rules  
2 10. 这是最关键的问题。基于操作规则和自我反思，下一步在有效操作集合中最合适的操作应是什么?

1 and self-reflection, what should be the most suitable action in  
2 你应逐步分析该操作的效果。

1 the valid action set for the next step? You should analyze the  
2 不应重复之前的操作，也不要尝试验证之前的操作是否成功。

1 effects of the action step by step. You should not repeat the

1 previous action again. Do not try to verify whether the previous

1 action succeeded.

1 11. Do all the selected actions exist in the valid action set? If  
2 11. 所有选定的动作都存在于有效动作集合中吗？如果

1 no, regenerate the action and give the reasons.  
2 不，重新生成动作并给出理由。

1 12. If you are placing a road, is the road more than 300 pixels  
2 12. 如果您正在铺设道路，道路是否超过300像素

1 long? Otherwise, regenerate the action and give reasons.  
2 <long?>否则，请重新生成该操作并说明理由。

Actions: The requirements that the generated action needs to follow.

操作：生成的操作需要遵循的要求。

1       The best action, or short sequence of actions without gaps, to  
2       最佳动作，或无间断的短动作序列，来

1       execute next to progress in achieving the goal. Pay attention to  
2       执行下一步以推进目标的实现。注意

1       the names of the available skills and to the previous skills  
2       可用技能的名称及之前的技能

1       already executed, if any. You should also pay more attention to  
2       已执行的（如有）。你还应更加注意

1       the following action rules:  
2       以下操作规则：

1       1. You should output actions in Python code format and specify any  
2       1. 你应以Python代码格式输出操作，并说明任何

1       necessary parameters to execute that action. If the function has  
2       执行该操作所需的必要参数。如果该函数具有

1       parameters, you should also include their names and decide their  
2       参数时，您还应包括它们的名称并确定它们的

1       values, like "move\_right(duration=1)". If it does not have a  
2       值，例如 "move\_right(duration=1)"。如果没有

1       parameter, just output the action, like "open\_map()".  
2       参数，只需输出动作，如"open\_map()"。

1       2. Given the current situation and task, you should only choose  
2       鉴于当前情况和任务，您应仅选择

1       the most suitable action from the valid action set. You cannot use  
2       从有效动作集合中选择最合适的作品。你不能使用

1       actions that are not in the valid action set to control the  
2       不在有效动作集合中的动作来控制

1       character.  
2       角色。

1       3. You MUST NOT output more than one skill in the actions.  
2       3. 你不得在动作中输出超过一个技能。

1       4. If you want to build a village, you should follow these rules:  
2       4. 如果你想建造村庄，应遵循以下规则：

1       4.1 Build roads correctly.  
2       4.1 正确建造道路。

1       - If you have not opened the road tool, you should open the menu  
2       - - 如果你尚未打开道路工具，应打开菜单

1       . If you have already opened the menu, you should not open it  
2       . 如果菜单已打开，则不应再次打开

1       again.  
2       .

1           - Newly built roads must be connected to the existing roads.  
2           - - 新建道路必须与现有道路相连。

1           - Determine in which grid the starting point of the newly built  
2           - - 确定新建道路起点所在的网格，并识别起点的像素位置。

1           road is located, and identify the pixel position of the starting

1           point.

1           - Build the road in the correct direction.  
2           - - 按正确方向建造道路。

1           5. You MUST NOT repeat the previous action with the same  
2           5. 如果你认为前一个动作失败，不能重复使用相同参数执行该动作。

1           parameters again if you think the previous action fails.

1           6. Your action should strictly follow the analysis in the  
2           6. 你的操作应严格遵循推理中的分析。

1           reasoning. Do not output any additional action not mentioned in  
2           不要输出推理中未提及的任何额外操作。

1           the reasoning.

1           7. Please do not directly connect the entrance of the highway with  
2           7. 请不要一开始就直接将高速公路入口与出口相连。

1           the exit of the highway at the beginning. To make the village as  
2           为了使村庄尽可能大，你应当在野外修建道路并将它们相互连接。

1           large as possible. You should build roads in the wild and connect

1           them with each other.

1           8. If you are placing a road, the road needs to be at least 300  
2           8. 如果你正在铺设道路，道路长度需至少为300像素。

1           pixels long.

You should only respond in the format described below, and you should  
你应仅以以下描述的格式回应，且不得输出评论或其他信息。

1           not output comments or other information.

Reasoning:

推理:

1. ...

2. ...

3. ...

...

Actions:

操作:

cccpython

cccpython

1 | action (args1=x,args2=y)  
2 | action (args1=x,args2=y)

ccc

ccc

---

Prompt 10: Stardew: Information Gathering Cultivation prompt.

提示10: 星露谷物语 (Stardew) : 信息收集培养提示。

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手

1 | on the PC, equipped to handle a wide range of tasks in the game.  
2 | 在电脑上，配备了处理游戏中各种任务的能力。

1 | Your advanced capabilities enable you to process and interpret  
2 | 您的高级能力使您能够处理和解读

1 | gameplay screenshots and other relevant information.  
2 | 游戏截图及其他相关信息。

<\$image\_introduction\$>

Current task:

当前任务：

<\$task\_description\$>

Description: Please analyze and describe the screenshot image in a

描述：请分析并描述截图图像中的内容

1 grid-by-grid format and then provide an overall image description.  
2 按网格格式逐格描述，然后提供整体图像描述。

1 Pay attention to anything related to the task. The image is  
2 注意与任务相关的任何内容。图像是

1 divided into a 3x5 grid, each cell having its own coordinates. For  
2 <divided into a 3x5 grid, each cell having its own coordinates. For</text> 分成3x5的网格，每  
个单元格都有自己的坐标。对于

1 each grid cell, describe the contents in detail, focusing on any  
2 每个网格单元，详细描述内容，重点关注任何

1 critical icons, or objects present in that particular segment. If  
2 关键图标，或该特定片段中出现的对象。如果

1 there are specific features such as characters or text, mention  
2 有特定的特征，如字符或文本，提及

1 these as well. After completing the description for one cell,  
2 这些也是。在完成对一个单元格的描述后，

1 proceed to the next, for example, 'In grid (1,1), [description].'  
2 继续到下一个，例如，“在网格 (1,1) 中，[描述]。”

1 In grid \$\left( \begin{matrix} 1,2 \end{matrix} \right)\$ , [description]. \${}^{\prime}\$ and so on until the entire  
image is  
2 在网格\$\left( \begin{matrix} 1,2 \end{matrix} \right)\$中，[描述]。\${}^{\prime}\$，如此类推，直到整个图像完成

1 covered.  
2 已覆盖。

Date\_time: The date and time information in the game are shown on the

日期时间：游戏中的日期和时间信息显示在

1 upper-right of the screenshot, in grid (1, 5). An example of the  
2 截图的右上角，位于网格(1, 5)处。日期和时间信息的示例是

1 date and time information is "Wed 10, 5:10 pm".  
2 "周三 10号，下午5:10"。

Energy: The current energy remains for the character doing actions.

能量：角色执行动作时当前剩余的能量。

1 The energy bar is shown on the bottom-right of the screenshot, in  
2 能量条显示在截图的右下角，位于

1 grid (3, 5). The full energy is 270. An example of the energy  
2 网格(3, 5)处。满能量为270。能量信息的示例是"150/270"。

1 information is " 150/270".

Weather: The current weather information in the game, the weather is

天气: 游戏中的当前天气信息, 天气类型包括“晴天”、“雨天”、“有风”、“下雪”、“暴风雨”、“节日”、“婚礼”和“无”。

1 | one from "Sunny", "Rainy", "Windy", "Snowy", "Stormy", "Festival",  
2 | 如果都不符合, 则只输出"无"。

1 | "Wedding", and "null". If none of them applies, only output "null"  
2 | 对话: 如果截图中显示有对话, 提取对话文本, 如"店主: 你想买什么?", 否则只输出"无"。

1 | ".

Dialog: If there are some dialogs shown in the screenshot, extract the

1 | text of the conversation, like "Shopkeeper: What do you want to

1 | buy?", otherwise, only output "null".

Other: Other information that does not belong to the above categories.

其他: 不属于上述类别的其他信息。

1 | If none of them applies, only output "null".  
2 | 如果都不符合, 则只输出"无"。

You should only respond in the format described below and not output

您应仅按照以下描述的格式回复, 不要输出

1 | comments or other information.  
2 | 评论或其他信息。

Description:

描述:

In grid (1,1) , ...

在网格(1,1)中, ...

In grid (1,2) , ...

在网格(1,2)中, ...

...

In grid (3,5) , ...

在网格(3,5)中, ...

Overall, the image shows...

整体来看, 图像显示了...

Date\_time:

日期时间:

Date and time information

日期和时间信息

Energy :

能量:

The number of energy remains showing in the energy bar

能量条中显示的能量数值

Weather :

天气:

Weather information

天气信息

Dialog :

对话:

Dialog text

对话文本

Other:

其他:

Other information is ...

其他信息是...

---

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手

1 | on the PC, equipped to handle a wide range of tasks in the game.  
2 | 运行于PC平台，能够处理游戏中的各种任务。

1 | Your advanced capabilities enable you to process and interpret  
2 | 你具备先进的能力，可以处理和解析

1 | gameplay screenshots and other relevant information. Your task is  
2 | 游戏截图及其他相关信息。你的任务是

1 | to examine these inputs, interpret the in-game context, and  
2 | 审查这些输入，解读游戏内的情境，

1 | determine whether the executed action takes effect.  
2 | 并判断所执行的操作是否生效。

Target task:

目标任务:

<\$task\_description\$>

Current subtask for completing the target task:

完成目标任务的当前子任务：

<\$subtask\_description\$>

The reasoning for proposing the current subtask:

提出当前子任务的理由：

<\$subtask\_reasoning\$>

Last executed action for completing the subtask:

完成子任务的最后执行操作：

<\$previous\_action\$>

Reasoning for the last action:

最后操作的理由：

<\$previous\_reasoning\$>

Current date and time:

当前日期和时间：

<\$date\_time\$>

Previous toolbar information:

之前的工具栏信息：

<\$previous\_toolbar\_information\$>

Current toolbar information:

当前的工具栏信息：

<\$toolbar\_information\$>

Summarization of recent history:

近期历史总结：

<\$history\_summary\$>

<\$image\_introduction\$>

Reasoning: You need to answer the following questions step by step to

推理：你需要逐步回答以下问题以

1 | get some reasoning based on the last action and sequential frames  
2 | 基于上一次动作和连续帧进行推理

1 | of the character during the execution of the last action.  
2 | 在执行最后一个动作时角色的状态。

1. What is the executed action? Please answer this question not based

2. 执行的操作是什么？请不要基于此问题作答

1 | on the sequential frames.  
2 | 在连续帧上。

2. Was the executed action successful? Give reasons. You should refer

3. 执行的操作成功了吗？请说明理由。你应当参考

1 | to the following rules:  
2 | 遵循以下规则：

- If the action involves moving forward, it is considered unsuccessful
- 如果动作涉及向前移动，则视为不成功

1 | only when the character's position remains unchanged across  
2 | 只有当角色的位置保持不变时

1 | sequential frames, regardless of background elements and other  
2 | 连续帧，无论背景元素及其他

1 | people.  
2 | 人们。

- If you are not 100% sure that the action fails, regard it as success
- 如果你不能百分之百确定该操作失败，就视其为成功

3. If the last action is not executed successfully, what is the most

4. 如果最后一个操作未成功执行，最重要的是

1 | probable cause? You should give only one cause and refer to the  
2 | 合理的理由？你只需给出一个原因并参考

1 | following rules:  
2 | 以下规则：

- The reasoning for the last action could be wrong.
  - 上一次操作的推理可能是错误的。
- 
- If it is an interaction action, the most probable cause was that the
  - 如果是交互动作，最可能的原因是

1 | action was unavailable at the current place, then you should move  
2 | 当前地点无法执行该操作，您应当移动位置

1 | to a new place.

2 | 到一个新地方。

- If it is a movement action, the most probable cause was that you
- 如果是移动动作，最可能的原因是你

1 | were blocked by seen or unseen obstacles.

2 | 被可见或不可见的障碍物阻挡。

- If there is an error report, analyze the cause based on the report.
- 如果有错误报告，请根据报告分析原因。

4. Is the subtask completed? Give your reasons. If you want to make

5. 子任务完成了吗？请说明理由。如果你想要

1 | any confirmation, regard it as a success.

2 | 任何确认，都视为成功。

5. Is the target task completed? Give your reasons.

6. 目标任务完成了吗？请说明理由。

6. Do you think the subtask is reasonable? Give your reasons.

7. 你认为这个子任务合理吗？请说明理由。

---

---

You should only respond in the format described below.

您应仅以以下描述的格式回复。

Reasoning:

推理：

1 | 1. ...

2. ...

1 | 3. ...

1 | ...

---

Prompt 12: Stardew: Task Inference Cultivation prompt.

提示12：星露谷物语（Stardew）：任务推断培养提示。

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手

1 | on the PC, equipped to handle a wide range of tasks in the game.  
2 | 在电脑上，配备了能够处理游戏中各种任务的功能。

1 | You will also be given a summary of the history that happened  
2 | 你还将获得一份发生过的历史摘要

1 | before the last screenshot. You should assist in summarizing the  
2 | 在最后一个截图之前。你应协助总结

1 | events for future decision-making and also propose a new subtask,  
2 | 事件以便未来决策，并提出一个新的子任务，

1 | which is the most suitable subtask for the current situation,  
2 | 即在当前情境下最合适的子任务，

1 | given the target task.  
2 | 基于目标任务。

Here is some helpful information to help you do the summarization and

以下是一些有助于你进行总结和

1 | propose the subtask.  
2 | 提出子任务的有用信息。

Current task:

当前任务：

<\$task\_description\$>

Previous proposed subtask for the task:

之前为该任务提出的子任务：

<\$subtask\_description\$>

Previous reasoning for proposing the subtask:

提出该子任务的先前推理：

<\$subtask\_reasoning\$>

<\$image\_introduction\$>

Current toolbar information:

当前工具栏信息：

<\$toolbar\_information\$>

Last executed action:

最后执行的操作：

<\$previous\_action\$>

Decision-making reasoning for the last executed action:

对最后执行操作的决策推理：

<\$previous\_reasoning\$>

Self-reflection for the last executed action:

对最后执行操作的自我反思：

<\$self\_reflection\_reasoning\$>

The following is the summary of history that happened before the last

以下是发生在最后一个

1 | screenshot:

2 | 截图之前的历史总结：

<\$previous\_summarization\$>

History\_summary: Summarize what happened in the past experience,

历史总结：总结过去经历中发生的事情，

1 | especially the last step according to the decision-making  
2 | 尤其是根据决策的最后一步

1 | reasoning and self-reflection reasoning for the last executed  
2 | 对最后执行的推理和自我反思推理

1 | action. The summarization needs to be precise, concrete and highly  
2 | 行动。总结需要精准、具体且高度

1 | related to the task and follow the rules below.  
2 | 与任务相关并遵守以下规则。

1. Summarize the tasks from the history and the current task. What is

2. 总结历史任务和当前任务。内容是什么

1 | the current progress of the task? For example, to harvest a seed,  
2 | 任务的当前进展如何？例如，收获种子时，

1 you need to water the seed for 4 days. And you have already  
2 你需要给种子浇水4天。你已经浇了

1 planted the seed and watered it for two days.  
2 种下种子并浇水两天。

2. Record the successful actions and organize them into events day by

3. 记录成功的操作并将其按天整理成事件

1 day.  
2 天。

3. Do not forget the information and key events in the previous days.

4. 不要忘记前几天的信息和关键事件。

4. If you are watering a seed. Record how many times you have watered

5. 如果你正在给种子浇水，记录浇水的次数

1 and calculate how many days you have to water before you can  
2 并计算你需要浇水多少天才能

1 harvest according to the toolbar information provided above.  
2 根据上述工具栏信息进行收割。

Here is an example to follow:

以下是一个示例：

---

---

On Thu.4, I dig the dirt with the toe and then plant the parsnip seed

在星期四，我用脚趾挖土，然后种下欧防风种子

1 and water the seed. The seed has been watered once. It still needs  
2 并浇水。种子已经浇过一次水了，仍然需要

1 to be watered another three times to harvest. On Fri.5, I watered  
2 还需浇水三次才能收获。5号星期五，我浇了水

1 the seed again. The seed has been watered twice. It still needs  
2 再次播种。种子已经浇了两次水，仍然需要

1 to be watered twice to harvest. Today, Sat.6, I just need to get  
2 浇水两次才能收获。今天，星期六6号，我只需要去拿

1 out of home and watered the seed again.  
2 走出家门，又给种子浇了水。

Subtask\_reasoning: Decide whether the previous subtask is finished and

子任务推理：判断前一个子任务是否已完成并

1 whether it is necessary to propose a new subtask. The subtask  
2 是否有必要提出一个新的子任务。该子任务

1 should be straightforward, contribute to the target task and be  
2 应该是直接明了的，有助于目标任务，并且

1 most suitable for the current situation, which should be completed  
2 最适合当前情况，应当完成

1 within a few actions. You should respond to me with:  
2 在几步操作内。你应该回复我：

1. How to finish the target task? You should analyze it step by step.

2. 如何完成目标任务？你应该逐步分析。

2. What is the current progress of the target task according to the

3. 根据当前情况，目标任务的进展如何

1 analysis in step 1? Please do not make any assumptions if they are  
2 步骤1中的分析？如果有，请不要做任何假设

1 not mentioned in the above information. You should assume that  
2 上述信息中未提及。您应当假设

1 you are doing the task from scratch.

2 你正在从头开始完成任务。

3. What is the previous subtask? Does the previous subtask finish? Or

4. 上一个子任务是什么？上一个子任务完成了吗？还是

1 is it improper for the current situation? Then select a new one,  
2 当前情况不合适吗？那就选择一个新的，

1 otherwise you should reuse the last subtask.

2 否则你应该重复使用上一个子任务。

4. If you want to propose a new subtask, give reasons why it is more

5. 如果你想提出一个新的子任务，请说明为什么它更

1 feasible for the current situation.

2 适合当前情况。

5. The proposed subtask needs to be precise and concrete within one

6. 所提议的子任务需要在一句话内准确具体。

1 sentence. It should not be related to any skills.

2 它不应涉及任何技能。

6. The seed only needs to be watered once.

7. 种子只需浇水一次。

7. Do not mention any grid information in the subtask description.

8. 子任务描述中不要提及任何网格信息。

8. Do not check the growth status of the crop.

9. 不要检查作物的生长状态。

9. The seeds only need to be watered ONCE every day. If you have

10. 种子每天只需浇水一次。如果你今天已经浇过水，

1 already watered the seed today, you should return home and go to  
2 你应该回家休息，等待第二天。

1 sleep, waiting for the next day.  
2 睡觉，等待下一天。

You should only respond in the format described below, and you should  
你应仅以以下描述的格式回复，且不应输出评论或其他信息。

1 not output comments or other information.  
2 不应输出评论或其他信息。

History\_summary :

历史摘要：

The summary is...

摘要是...

Subtask\_reasoning:

子任务推理：

1. ...

2. ...

...

Subtask:

子任务：

The current subtask is

当前子任务是

---

Prompt 13: Stardew: Action Planning Cultivation prompt.

提示13：星露谷物语：行动规划培养提示。

---

You are a helpful AI assistant integrated with 'Stardew Valley' on the  
你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手

1 PC, equipped to handle various tasks in the game. Your advanced  
2 配备多功能处理能力的电脑，能够应对游戏中的各种任务。您的先进

1 capabilities enable you to process and interpret gameplay  
2 功能使您能够处理和解读游戏玩法

1 | screenshots and other relevant information. By analyzing these  
2 | 截图及其他相关信息。通过分析这些

1 | inputs, you gain a comprehensive understanding of the current  
2 | <inputs, you gain a comprehensive understanding of the current>

输入，您可以全面了解当前情况

1 | context and situation within the game. Utilizing this insight, you  
2 | 游戏中的背景和情境。利用这一洞察，您

1 | are tasked with identifying the most suitable in-game action to  
2 | 负责识别最合适的游戏内动作以

1 | take next, given the current task. You control the game character  
2 | 根据当前任务采取下一步行动。你控制游戏角色

1 | and can execute actions from the available action set. Upon  
2 | 并且可以从可用的动作集合中执行动作。

1 | evaluating the provided information, your role is to articulate  
2 | 评估所提供的信息，您的职责是阐述

1 | the precise action you would deploy, considering the game's  
2 | 考虑到游戏的情况，你将采取的具体行动

1 | present circumstances, and specify any necessary parameters for  
2 | 当前情况，并指定任何必要的参数以便

1 | implementing that action.  
2 | 执行该操作。

Here is some helpful information to help you make the decision.

这里有一些有助于你做出决定的信息。

Current subtask:

当前子任务：

<\$subtask\_description\$>

---

Current date and time:

当前日期和时间：

<\$date\_time\$>

---

---

Toolbar information:

工具栏信息：

<\$toolbar\_information\$>

Last executed action:

上次执行的操作:

<\$previous\_action\$>

Reasoning for the last action:

上次操作的推理:

<\$previous\_reasoning\$>

Self-reflection for the last executed action:

对上次执行操作的自我反思:

<\$previous\_self\_reflection\_reasoning\$>

Summarization of recent history:

近期历史总结:

<\$history\_summary\$>

Valid action set in Python format to select the next action:

用于选择下一步操作的Python格式有效操作集:

<\$skill\_library\$>

<\$image\_introduction\$>

Based on the above information, analyze the current situation and

基于以上信息，分析当前情况并

1 | provide the reasoning for what you should do for the next step to  
2 | 提供你下一步应采取行动的推理，以

1 | complete the task. Then, you should output the exact action you  
2 | 完成任务。然后，你应输出你

1 | want to execute in the game. You should respond to me with:  
2 | 想在游戏中执行的具体操作。你应回复我：

Reasoning: You should think step by step and provide detailed

推理：你应逐步思考并提供详细的

1 reasoning to determine the next action executed on the current  
2 推理，以确定当前应执行的下一步操作。

1 state of the task. You need to answer the following questions step  
2 任务状态。您需要逐步回答以下问题

1 by step. You cannot miss the last question:  
2 逐步进行。你绝不能错过最后一个问题：

1 1. Analyze the information in the toolbar. Does it contain all the  
2 1. 分析工具栏中的信息。它是否包含所有

1 necessary items for completing the task?  
2 完成任务所需的物品？

1 2. What is the current selected tool? Do you want to use a tool,  
2 2. 当前选中的工具是什么？您想使用某个工具吗，

1 such as axe, hoe, watering can, pickaxe and scythe? And is the  
2 例如斧头、锄头、洒水壶、镐和镰刀？还有

1 character's current position a suitable place to use such a tool?  
2 角色当前的位置适合使用这种工具吗？

1 Then you should use use\_tool() instead of do\_action().  
2 那么你应该使用 use\_tool() 而不是 do\_action()。

1 3. Does the character already reach the target place?  
2 3. 角色已经到达目标地点了吗？

1 4. What was the previous action? If the previous action was a  
2 4. 上一个动作是什么？如果上一个动作是

1 movement, were you blocked?  
2 移动时，你被阻挡了吗？

1 5. If your task is to harvest the plant, did you water the seed?  
2 5. 如果你的任务是收获植物，你给种子浇水了吗？

1 The seeds only need to be watered ONCE every day. If you have  
2 种子每天只需浇水一次。如果你有

1 already watered the seed today, you should return home and go to  
2 今天已经给种子浇过水了，你应该回家去

1 sleep, waiting for the next day.  
2 睡觉，等待新的一天。

1 6. This is the most critical question. Based on the action rules  
2 6. 这是最关键的问题。基于行动规则

1 and self-reflection, what should be the most suitable action in  
2 经过自我反思，下一步在有效动作集合中最合适的动作应该是什么？

1 the valid action set for the next step? You should analyze the  
2 你应当逐步分析该动作的效果。

1 effects of the action step by step. You should not repeat the  
2 除非是移动动作，否则不应重复之前的动作。

1 previous action again except for the movement action. Do not try  
2 不要尝试验证之前的动作是否成功。

1 | to verify whether the previous action succeeded.

1 | 7. Is the selected action the same as the last executed action? If  
2 | 7. 选定的动作是否与上一次执行的动作相同？如果是，重新生成动作并说明理由。

1 | yes, regenerate the action and give the reasons.

1 | 8. Do all the selected actions exist in the valid action set? If  
2 | 8. 所有选定的动作是否都存在于有效动作集合中？如果否，重新生成动作并说明理由。

1 | no, regenerate the action and give the reasons.

1 | 9. Analyze whether the selected action meets the requirements of  
2 | 9. 逐一分析选定的动作是否满足以下动作 (Actions) 的要求。生成的动作是否满足所有要求？如果不满足，  
重新生成动作并说明理由。

1 | the Actions below one by one. Does the generated action meet all

1 | the requirements? If not, regenerate the action and give the

1 | reasons.

Actions: The requirements that the generated action needs to follow.

动作 (Actions) : 生成动作需要遵循的要求。

1 | The best action, or short sequence of actions without gaps, to  
2 | 接下来执行的最佳动作，或无间断的短动作序列，以推动目标的实现。

1 | execute next to progress in achieving the goal. Pay attention to  
2 | 注意

1 | the names of the available skills and to the previous skills  
2 | 可用技能的名称以及之前的技能

1 | already executed, if any. You should also pay more attention to  
2 | 已执行的（如有）。您还应更加注意

1 | the following action rules:

2 | 以下操作规则：

---

1 | 1. You should output actions in Python code format and specify any  
2 | 1. 你应以Python代码格式输出操作，并说明任何

1 necessary parameters to execute that action. If the function has  
2 执行该操作所需的参数。如果函数具有

1 parameters, you should also include their names and decide their  
2 参数时，您还应包括它们的名称并确定它们的

1 values, like "move\_right(duration=1)". If it does not have a  
2 值，比如 "move\_right(duration=1)"。如果它没有

1 parameter, just output the action, like "open\_map()".  
2 参数，只需输出动作，如"open\_map()"。

1 2. You can only output at most two actions in the output.  
2 2. 你最多只能输出两个操作。

1 3. In the screenshots, the blue band represents the left side and  
2 3. 在截图中，蓝色带表示左侧，且

1 the yellow band represents the right side. Please ignore character  
2 黄色带表示右侧。请忽略字符

1 's facing direction and output the action in an absolute direction  
2 面向方向并以绝对方向输出动作

1 like right and left.  
2 像左右一样。

1 4. If you want to interact with the objects in the toolbar, you  
2 4. 如果你想与工具栏中的对象进行交互，

1 need to make sure that the target object is already selected. You  
2 需要确保目标对象已被选中。你

1 need to use select\_tool() to select them before executing use\_tool  
2 需要先使用select\_tool()选择它们，然后才能执行use\_tool

1 () or do\_action().  
2 () 或 do\_action()。

1 5. If you want to plant a seed or harvest a mature crop, please  
2 5. 如果你想种下种子或收获成熟的作物，请

1 use do\_action() instead of use\_tool(). If you want to use tools,  
2 使用 do\_action() 而不是 use\_tool()。如果你想使用工具，

1 like axe, hoe, watering can, pickaxe and scythe, please use  
2 比如斧头、锄头、水壶、镐和镰刀，请使用

1 use\_tool().  
2 use\_tool()。

1 6. If upon self-reflection you think the last action was  
2 6. 如果经过自我反思你认为上一个动作在当前位置不可用，

1 unavailable at the current place, you MUST move to another place.  
2 你必须移动到另一个地方。

1 Please do not try to execute the same action again.  
2 请不要尝试再次执行相同动作。

1 7. If you want to get out of the house, just use the skill  
2 7. 如果你想离开房子，只需使用技能

1 | get\_out\_of\_house(). You MUST NOT output any movement action behind  
2 | get\_out\_of\_house()。你绝对不能在此技能后输出任何移动动作。

1 | this skill. And if the last executed action already contains this  
2 | 如果上一个执行的动作已经包含此技能，当前步骤不要再次执行该技能。

1 | skill, do not execute this skill for the current step again.

1 | 8. If upon self-reflection you think you were blocked, you MUST  
2 | 8. 如果经过自我反思你认为被阻挡了，必须

1 | change the direction of moving, so that you can pass obstacles.  
2 | 改变移动方向，以便绕过障碍物。

1 | 9. You MUST NOT repeat the previous action again if you think the  
2 | 9. 如果你认为上一个动作失败，绝对不能重复执行该动作。

1 | previous action fails.

1 | 10. Your action should strictly follow the analysis in the  
2 | 10. 你的行动应严格遵循推理中的分析

1 | reasoning. Do not output any additional action not mentioned in  
2 | 。不要执行推理中未提及的任何额外操作

1 | the reasoning.

2 | 。

You should only respond in the format described below, and you should

你应仅以以下描述的格式回复，且不应输出评论或其他信息

1 | not output comments or other information.  
2 | 。

Reasoning :

推理：

1. ...

2. ...

3. ...

Actions:

操作：

44 CPython

44 CPython

1 | action ( \${args1} = x,{args2} = y\$ )  
2 | 操作 ( \${args1} = x,{args2} = y\$ )

ccc

ccc

---

Prompt 14: Stardew: Information Gathering Farm Clearup prompt.

提示14: Stardew (星露谷物语) : 信息收集农场清理提示

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是集成在PC版《星露谷物语》中的一个有用的AI助手

1 | on the PC, equipped to handle a wide range of tasks in the game.  
2 | , 能够处理游戏中的各种任务

1 | Your advanced capabilities enable you to process and interpret  
2 | 。你具备先进的能力，能够处理和解读

1 | gameplay screenshots and other relevant information.  
2 | 游戏截图及其他相关信息

<\$image\_introduction\$>

Current task:

当前任务：

<\$task\_description\$>

Description: Please analyze and describe the screenshot image in a

描述：请分析并描述截图图像中的内容

1 | grid-by-grid format and then provide an overall image description.  
2 | 按网格格式逐格描述，然后提供整体图像描述。

1 | Pay attention to anything related to the task. The image is  
2 | 注意与任务相关的任何内容。图像是

1 | divided into a 3x5 grid, each cell having its own coordinates. For  
2 | <divided into a 3x5 grid, each cell having its own coordinates. For</text>被翻译为：划分为3x5  
的网格，每个单元格都有自己的坐标。对于

1 | each grid cell, describe the contents in detail, focusing on any  
2 | 每个网格单元，详细描述内容，重点关注任何

1 | critical icons, or objects present in that particular segment. If  
2 | 关键图标，或该特定片段中出现的对象。如果

1 | there are specific features such as characters or text, mention  
2 | 有特定的特征，如字符或文本，提及

1 these as well. After completing the description for one cell,  
2 这些也是。在完成对一个单元格的描述后，  
1 proceed to the next, for example, 'In grid (1,1), [description].  
2 继续到下一个，例如，"在网格 (1,1) 中，[描述]。"

---

1 In grid \$\left( \begin{matrix} 1,2 \end{matrix} \right)\$ , [description]. \${}^{\prime}\$ and so on until the entire  
image is  
2 在网格\$\left( \begin{matrix} 1,2 \end{matrix} \right)\$, [描述]。\${}^{\prime}\$, 依此类推，直到整个图像完成  
1 covered.  
2 覆盖。

Date\_time: The date and time information in the game are shown on the  
日期时间：游戏中的日期和时间信息显示在

1 upper-right of the screenshot, in grid (1, 5). An example of the  
2 截图的右上角，位于网格(1, 5)中。一个示例  
1 date and time information is "Wed 10, 5:10 pm".  
2 日期和时间信息是"周三10号，下午5:10"。

Energy: The current energy remains for the character doing actions.  
能量：当前能量保持不变，适用于执行动作的角色。

1 The energy bar is shown on the bottom-right of the screenshot, in  
2 能量条显示在截图的右下角，位于  
1 grid (3, 5). The full energy is 270. An example of the energy  
2 网格 (3, 5)。总能量为270。能量的一个示例  
1 information is " 150/270".  
2 信息为"150/270"。

Weather: The current weather information in the game, the weather is  
天气：游戏中的当前天气信息，天气为

1 one from "Sunny", "Rainy", "Windy", "Snowy", "Stormy", "Festival",  
2 "晴朗"、"雨天"、"有风"、"下雪"、"暴风雨"、"节日"、  
1 "Wedding", and "null". If none of them applies, only output "null"  
2 "婚礼"，或"无"。如果都不符合，则仅输出"无"。  
1 ".

Dialog: If there are some dialogs shown in the screenshot, extract the  
对话：如果截图中显示有对话，提取

1 text of the conversation, like "Shopkeeper: What do you want to  
2 对话文本，如"店主：你想买什么？"，否则仅输出"无"。

1 | buy?", otherwise, only output "null".

Other: Other information that does not belong to the above categories.

其他: 不属于上述类别的其他信息。

1 | If none of them applies, only output "null".

2 | 如果都不符合, 则仅输出"无"。

You should only respond in the format described below and not output

你应仅以以下描述的格式回复, 不要输出

1 | comments or other information.

2 | 评论或其他信息。

Description:

描述:

In grid (1,1) , ...

在网格 (1,1) , ...

In grid (1,2) , ...

在网格 (1,2) , ...

...

In grid (3,5) , ...

在网格 (3,5) , ...

Overall, the image shows...

总体来看, 图像显示了...

Date\_time:

日期时间:

Date and time information

日期和时间信息

Energy:

能量:

The number of energy remains showing in the energy bar

能量条中显示的能量数值

Weather :

天气:

Weather information

天气信息

Dialog :

对话:

Dialog text

对话文本

Other:

其他:

Other information is ...

其他信息是...

---

Prompt 15: Stardew: Self-Reflection Farm Clearup prompt.

提示15：星露谷物语（Stardew）：自我反思农场清理提示。

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是集成在“星露谷物语”（Stardew Valley）中的一个有用的AI助手

1 | on the PC, equipped to handle a wide range of tasks in the game.

2 | 运行于PC平台，能够处理游戏中的各种任务。

1 | Your advanced capabilities enable you to process and interpret

2 | 你具备先进的能力，能够处理和解读

1 | gameplay screenshots and other relevant information. Your task is

2 | 游戏截图及其他相关信息。你的任务是

1 | to examine these inputs, interpret the in-game context, and

2 | 检查这些输入，解读游戏内的上下文，

1 | determine whether the executed action takes effect.

2 | 并确定所执行的动作是否生效。

Target task:

目标任务：

<\$task\_description\$>

Current subtask for completing the target task:

完成目标任务的当前子任务：

<\$subtask\_description\$>

The reasoning for proposing the current subtask:

提出当前子任务的推理依据：

<\$subtask\_reasoning\$>

Last executed action for completing the subtask:

完成子任务的上一次执行动作：

<\$previous\_action\$>

Reasoning for the last action:

上一次动作的推理依据：

---

---

<\$previous\_reasoning\$>

Current date and time:

当前日期和时间：

<\$date\_time\$>

Previous toolbar information:

之前的工具栏信息：

<\$previous\_toolbar\_information\$>

Current toolbar information:

当前的工具栏信息：

<\$toolbar\_information\$>

Summarization of recent history:

近期历史总结：

<\$history\_summary\$>

<\$image\_introduction\$>

Reasoning: You need to answer the following questions step by step to

推理：你需要逐步回答以下问题，

- 1 | get some reasoning based on the last action and sequential frames
- 2 | 基于上一次动作和角色执行该动作时的连续帧，  
1 | of the character during the execution of the last action.  
2 | 进行一些推理。

1. What is the executed action? Please answer this question not based
2. 执行的动作是什么？请不要基于连续帧回答此问题。

1 | on the sequential frames.

2. Was the executed action successful? Give reasons. You should refer

3. 执行的动作成功了吗？请给出理由。你应参考

1 | to the following rules:

2 | 以下规则：

- If the action involves moving forward, it is considered unsuccessful
- 如果动作涉及向前移动，只有当角色的位置在连续帧中保持不变时，才视为失败，

1 | only when the character's position remains unchanged across

2 | 无论背景元素和其他人物如何。

1 | sequential frames, regardless of background elements and other

1 | people.

- If you are not 100% sure that the action fails, regard it as success

- 如果你不100%确定动作失败，则视为成功。

3. If the last action is not executed successfully, what is the most

4. 如果最后一个动作未成功执行，最可能的原因是什么？你应只给出一个原因，并参考

1 | probable cause? You should give only one cause and refer to the

2 | 以下规则：

1 | following rules:

- The reasoning for the last action could be wrong.

- 最后一个动作的推理可能是错误的。

- If it is an interaction action, the most probable cause was that the

- 如果是交互动作，最可能的原因是该动作在当前位置不可用，那么你应移动

1 | action was unavailable at the current place, then you should move

2 | 到新的位置。

1 | to a new place.

- If it is a movement action, the most probable cause was that you

- 如果是移动动作，最可能的原因是你被可见或不可见的障碍物阻挡。

1 | were blocked by seen or unseen obstacles.

- If there is an error report, analyze the cause based on the report.
  - 如果有错误报告，请根据报告分析原因。
4. Is the subtask completed? Give your reasons. If you want to make  
5. 子任务完成了吗？请说明理由。如果需要确认，视为成功。

1 | any confirmation, regard it as a success.  
2 | 任何确认，都视为成功。

5. Is the target task completed? Give your reasons.  
6. 目标任务完成了吗？请说明理由。  
6. Do you think the subtask is reasonable? Give your reasons.  
7. 你认为子任务合理吗？请说明理由。

You should only respond in the format as described below.  
你应仅以以下描述的格式回复。

Reasoning:

推理：

1. ...

2. ...

3. ...

...

---

Prompt 16: Stardew: Task Inference Farm Clearup prompt.

提示16：星露谷物语（Stardew Valley）：任务推断农场清理提示。

---

Assume you are a helpful AI assistant integrated with 'Stardew Valley'

假设你是集成于PC版《星露谷物语》中的有用AI助手，

- 1 | on the PC, equipped to handle a wide range of tasks in the game.  
2 | 能够处理游戏中的各种任务。
- 1 | You will also be given a summary of the history that happened  
2 | 你还将获得最后一张截图之前发生的历史摘要。
- 1 | before the last screenshot. You should assist in summarizing the  
2 | 你应协助总结事件以便未来决策，

1 events for future decision-making and also propose a new subtask,  
2 并提出一个新的子任务，

1 which is the most suitable subtask for the current situation,  
2 这是当前情况下最合适的子任务，

1 given the target task.  
2 基于目标任务。

Here is some helpful information to help you do the summarization and  
这里有一些有助于你进行总结和

1 propose the subtask.  
2 提出子任务。

Current task:

当前任务：

<\$task\_description\$>

Previous proposed subtask for the task:

之前为该任务提出的子任务：

<\$subtask\_description\$>

Previous reasoning for proposing the subtask:

提出子任务的先前推理：

<\$subtask\_reasoning\$>

<\$image\_introduction\$>

Current toolbar information:

当前工具栏信息：

<\$toolbar\_information\$>

Last executed action:

上一次执行的操作：

<\$previous\_action\$>

Decision-making reasoning for the last executed action:

上一次执行操作的决策推理：

<\$previous\_reasoning\$>

Self-reflection for the last executed action:

上一次执行操作的自我反思：

<\$self\_reflection\_reasoning\$>

The following is the summary of history that happened before the last

以下是上一次之前发生的历史总结

1 | screenshot :  
2 | 截图：

<\$previous\_summarization\$>

History\_summary: Summarize what happened in the past experience,

历史总结：总结过去的经历，

1 | especially the last step according to the decision-making  
2 | 特别是根据决策推理的最后一步  
  
1 | reasoning and self-reflection reasoning for the last executed  
2 | 和上一次执行操作的自我反思推理。  
  
1 | action. The summarization needs to be precise, concrete and highly  
2 | 总结需要准确、具体，且与任务高度相关，  
  
1 | related to the task and follow the rules below.  
2 | 并遵循以下规则。

1. Summarize the tasks from the history and the current task. What is

2. 总结历史中的任务和当前任务。内容应为

1 | the current progress of the task? For example, to harvest a seed,  
2 | 当前任务的进展如何？例如，收获种子时，  
  
1 | you need to water the seed for 4 days. And you have already  
2 | 你需要给种子浇水4天。你已经浇了  
  
1 | planted the seed and watered it for two days.  
2 | 种下种子并浇水两天。

2. Record the successful actions and organize them into events day by

3. 记录成功的操作并按天将其组织成事件

1 | day.  
2 | 天。  
  
3. Do not forget the information and key events in the previous days.  
4. 不要忘记前几天的信息和关键事件。  
  
4. If you are watering a seed. Record how many times you have watered  
5. 如果你正在给种子浇水，记录浇水的次数

1 and calculate how many days you have to water before you can  
2 并计算在你可以之前需要浇水多少天

1 harvest according to the toolbar information provided above.  
2 根据上述工具栏信息进行收割。

Here is an example to follow:

以下是一个示例：

On Thu.4, I dig the dirt with the toe and then plant the parsnip seed

在星期四，我用脚尖挖土，然后种下欧防风种子

1 and water the seed. The seed has been watered once. It still needs  
2 并浇水。种子已经浇过一次水了，仍然需要

1 to be watered another three times to harvest. On Fri.5, I watered  
2 还需浇水三次才能收获。5号星期五，我浇了水

1 the seed again. The seed has been watered twice. It still needs  
2 再次播种。种子已经浇了两次水，仍然需要

1 to be watered twice to harvest. Today, Sat.6, I just need to get  
2 需浇水两次才能收获。今天，星期六6号，我只需去取

1 out of home and watered the seed again.  
2 走出家门，又给种子浇了水。

Subtask\_reasoning: Decide whether the previous subtask is finished and

子任务推理：判断前一个子任务是否已完成并

1 whether it is necessary to propose a new subtask. The subtask  
2 是否有必要提出一个新的子任务。该子任务

1 should be straightforward, contribute to the target task and be  
2 应该直截了当，有助于目标任务，并且是

1 most suitable for the current situation, which should be completed  
2 最适合当前情况，应当完成

1 within a few actions. You should respond to me with:  
2 在几步操作内。你应该回复我：

1. How to finish the target task? You should analyze it step by step.

2. 如何完成目标任务？你应该逐步分析。

2. What is the current progress of the target task according to the

3. 根据当前情况，目标任务的进展如何

1 analysis in step 1? Please do not make any assumptions if they are  
2 步骤1中的分析？如果有，请不要做任何假设

1 not mentioned in the above information. You should assume that  
2 上述信息中未提及。您应当假设

1 you are doing the task from scratch.

2 你正在从头开始完成这项任务。

3. What is the previous subtask? Does the previous subtask finish? Or

4. 上一个子任务是什么？上一个子任务完成了吗？还是

1 is it improper for the current situation? Then select a new one,  
2 当前情况不合适吗？那就选择一个新的吧，

1 otherwise you should reuse the last subtask.  
2 否则你应该重用最后一个子任务。

4. If you want to propose a new subtask, give reasons why it is more

5. 如果您想提出一个新的子任务，请说明其更优的理由

1 feasible for the current situation.

2 适合当前情况。

---

5. The proposed subtask needs to be precise and concrete within one

6. 所提出的子任务需要在一个范围内明确且具体

1 sentence. It should not be related to any skills.

2 句子。它不应与任何技能相关。

6. The seed only needs to be watered once.

7. 种子只需浇水一次。

7. Do not mention any grid information in the subtask description.

8. 子任务描述中不要提及任何格子信息。

8. Do not check the growth status of the crop.

9. 不要检查作物的生长状态。

9. The seeds only need to be watered ONCE every day. If you have

10. 种子每天只需浇水一次。如果你已经

1 already watered the seed today, you should return home and go to  
2 今天浇过水了，你应该回家休息，

1 sleep, waiting for the next day.  
2 等待第二天。

You should only respond in the format described below, and you should

你应仅以下面描述的格式回复，且

1 not output comments or other information.  
2 不要输出评论或其他信息。

History\_summary:

历史摘要：

The summary is...

摘要是.....

Subtask\_reasoning:

子任务推理：

1. ...

2. ...

...

Subtask:

子任务:

The current subtask is

当前子任务是

---

Prompt 17: Stardew: Action Planning Farm Cleanup prompt.

提示17：星露谷物语（Stardew）：行动规划农场清理提示。

---

You are a helpful AI assistant integrated with 'Stardew Valley' on the

你是一个集成在“星露谷物语”（Stardew Valley）中的有用AI助手，

1 | PC, equipped to handle various tasks in the game. Your advanced  
2 | 配备了处理游戏中各种任务的个人电脑。您的先进

1 | capabilities enable you to process and interpret gameplay  
2 | 功能使您能够处理和解读游戏玩法

1 | screenshots and other relevant information. By analyzing these  
2 | 截图及其他相关信息。通过分析这些

1 | inputs, you gain a comprehensive understanding of the current  
2 | <inputs, you gain a comprehensive understanding of the current> 输入，您可以全面了解当前情况

1 | context and situation within the game. Utilizing this insight, you  
2 | <context>游戏中的背景和情境。利用这一洞察，您</context>

1 | are tasked with identifying the most suitable in-game action to  
2 | 负责识别最合适的游戏内动作以

1 | take next, given the current task. You control the game character  
2 | 根据当前任务，采取下一步行动。你控制游戏角色

1 | and can execute actions from the available action set. Upon  
2 | 并且可以从可用的动作集中执行动作。接着

1 | evaluating the provided information, your role is to articulate  
2 | 评估所提供的信息，您的职责是阐述

1 | the precise action you would deploy, considering the game's  
2 | 考虑到游戏的具体情况，你将采取的精确行动

1 | present circumstances, and specify any necessary parameters for  
2 | 当前情况，并指定任何必要的参数以便

1 | implementing that action.  
2 | 执行该操作。

Here is some helpful information to help you make the decision.

这里有一些有助于您做出决定的有用信息。

Current subtask:

当前子任务：

<\$subtask\_description\$>

Current date and time:

当前日期和时间：

<\$date\_time\$>

Toolbar information:

工具栏信息：

<\$toolbar\_information\$>

Last executed action:

上次执行的操作：

<\$previous\_action\$>

Reasoning for the last action:

上一次操作的推理：

<\$previous\_reasoning\$>

Self-reflection for the last executed action:

对上一次执行操作的自我反思：

<\$previous\_self\_reflection\_reasoning\$>

Summarization of recent history:

近期历史总结：

<\$history\_summary\$>

Valid action set in Python format to select the next action:

以Python格式表示的有效动作集，用于选择下一步动作：

<\$skill\_library\$>

<\$image\_introduction\$>

---

---

Based on the above information, analyze the current situation and

基于以上信息，分析当前状况并

1 | provide the reasoning for what you should do for the next step to  
2 | 提供你下一步应该做什么的理由

1 | complete the task. Then, you should output the exact action you  
2 | 完成任务。然后，你应输出确切的操作

1 | want to execute in the game. You should respond to me with:  
2 | 想在游戏中执行。你应该回复我：

Reasoning: You should think step by step and provide detailed

推理：你应该逐步思考并提供详细的

1 | reasoning to determine the next action executed on the current  
2 | 推理以确定当前执行的下一步操作

1 | state of the task. You need to answer the following questions step  
2 | 任务的状态。您需要逐步回答以下问题

1 | by step. You MUST NOT miss question 3 and question 11:  
2 | 一步步来。你绝对不能漏掉第3题和第11题：

1 | 1. Analyze the information in the tool bar. Does it contain all  
2 | 1. 分析工具栏中的信息。它是否包含所有内容

1 | the necessary items for completing the task?  
2 | 完成任务所需的物品？

1 | 2. Where is the character in the screenshot of the current step?  
2 | 2. 当前步骤截图中的角色在哪里？

1 | Where is the house in the screenshot of the current step? The blue  
2 | 当前步骤截图中的房子在哪里？蓝色的

1 | band represents the left side and the yellow band represents the  
2 | 红色带表示左侧，黄色带表示

1 right side. Where is the character compared with the house? (Is he  
2 右侧。这个角色相对于房子在哪里？(他是否

1 at the left edge or right edge of the house?)  
2 在房子的左边缘还是右边缘？)

1 3. If your task is to clear obstacles, you MUST NOT miss any  
2 3. 如果你的任务是清除障碍，绝对不能遗漏任何一个

1 question in this step:

2 此步骤中的问题：

1 - The blue band represents the left side and the yellow band  
2 - 蓝色带表示左侧，黄色带

1 represents the right side. Where is the character according to the  
2 表示右侧。根据

1 house? (Is he at the left edge or right edge of the house?)  
2 房子？(他是在房子的左边缘还是右边缘？)

1 - Which grids do the house span in the screenshot? (You MUST  
2 - 截图中房屋跨越了哪些网格？(您必须

1 answer one or two grid position. The house does not span over two  
2 回答一个或两个格子位置。房屋不会跨越两个格子

1 grids.) Then, what are the two grids below and near the house? (e.  
2 网格。)那么，下面靠近房子的两个网格是什么？(例如。

1 g. If the house spans from grid (1,3) to (1,4), the CLEARING AREA  
2 g. 如果房屋跨越网格(1,3)到(1,4)，则清理区域

1 of character should be grid  $\left( \begin{array}{l} 2 \\ 3 \end{array} \right)$  and  $\left( \begin{array}{l} 2 \\ 4 \end{array} \right)$ . If the house  
spans

2 字符的网格应为 $\left( \begin{array}{l} 2 \\ 3 \end{array} \right)$ 和 $\left( \begin{array}{l} 2 \\ 4 \end{array} \right)$ 。如果房屋跨越

1 grid  $\left( \begin{array}{l} 1 \\ 3 \end{array} \right)$ , the CLEARING AREA of character should be grid  $\left( \begin{array}{l} 2 \\ 2 \end{array} \right)$

2 网格 $\left( \begin{array}{l} 1 \\ 3 \end{array} \right)$ , 角色的清理区域应为网格 $\left( \begin{array}{l} 2 \\ 2 \end{array} \right)$

1 and  $\left( \begin{array}{l} 2 \\ 3 \end{array} \right)$ . You MUST remember this CLEARING AREA precisely IN THIS  
2 并且  $\left( \begin{array}{l} 2 \\ 3 \end{array} \right)$ 。你必须准确记住此清理区域

1 ROUND.) You should focus on obstacles in them. You MUST NOT move  
2 回合。)你应该关注其中的障碍。你绝对不能移动

1 the character out of these two obstacle grids.

2 从这两个障碍网格中移除该角色。

1 - In order to clear all obstacles below the house and make the  
2 - - 为了清除房屋下方的所有障碍物并使

1 place suitable for cultivating, you should not target for a  
2 适合耕作的地方，不应以此为目标

1 specific obstacle. Instead, you should try your best to move the  
2 具体障碍。相反，你应该尽力移动

1 character to pass every patch in the CLEARING AREA. You should  
2 角色需通过清理区域的每个补丁。你应该

1 | clear every obstacle that blocks the character in this process.  
2 | 清除阻碍角色前进的所有障碍。

1 |     - Every time after you move the character down (or up when being  
2 |     - 每次你将角色向下移动后 (或在被

1 |       too far from the house), you should move the character right or  
2 |       距离房子太远时, 你应该将角色向右移动或

1 | left (based on the character's position in the CLEARING AREA  
2 | 左侧 (基于角色在清理区域的位置)

1 |       compared with the house) to fully explore the CLEARING AREA of the  
2 |       相比房屋) 充分开发该清理区域 (CLEARING AREA)

1 | two grids determined above. You should clear all obstacles the  
2 | 上述确定的两个网格。你应该清除所有障碍物

1 | character meets in this process.  
2 | 角色在此过程中相遇。

1 |     - Is the current row fully explored by the character? If so,  
2 |     - 当前行是否已被角色完全探索? 如果是,

1 | your movement should be moving down. If there is an obstacle  
2 | 你的动作应该向下移动。如果有障碍物

1 | beneath the character, you should clear it first before moving the  
2 | 在字符下方, 移动之前应先清除它

1 | character down.  
2 | 角色倒下。

1 |     - You should not move too far from the house. You should not  
2 |     - 你不应该离房子太远。你不应该

1 | move the character down but should move him up instead if the  
2 | 向下移动角色, 但如果是则应向上移动他

1 | house is not in the current screenshot.  
2 | 房子不在当前截图中。

1 |     - What was the previous action? If the previous action contained  
2 |     - 之前的操作是什么? 如果之前的操作包含

1 | use\_tool(), you MUST NOT start with the same use\_tool() action in  
2 | use\_tool(), 你绝对不能以相同的 use\_tool() 操作开始

1 | this round. (You can still use use\_tool() by following a movement  
2 | 这一轮。 (你仍然可以通过跟随一个动作来使用use\_tool())

1 | or select\_tool()).  
2 | 或 select\_tool()).

1 |     - If the previous action was a movement, is the position of  
2 |     - - 如果前一个动作是移动, 当前位置是否为

1 | character changed? If not, it is the most trustworthy evidence  
2 | 字符改变了吗? 如果没有, 那就是最可靠的证据

1 |       that there is an obstacle in front of the character that can  
2 |       角色面前有一个障碍物, 可能会

1 interact with.  
2 互动。

1 - If the character is blocked by an obstacle in front of him or  
2 - 如果角色前方被障碍物阻挡，或者

1 if you think there is an obstacle in front of the character, what  
2 如果你认为角色面前有障碍物，怎么办

1 type of obstacle is it? (Usually, weed and grass are green, stone  
2 这是什么类型的障碍？（通常，杂草和草是绿色的，石头

1 is grey and branch is brown) What is the suitable tool for  
2 是灰色的，树枝是棕色的）适合的工具是什么

1 clearing it and is the tool correctly selected?  
2 清除它，并且工具是否正确选择？

1 4. What is the current selected tool? Do you want to use a tool,  
2 4. 当前选中的工具是什么？您想使用某个工具吗，

1 such as axe, hoe, watering can, pickaxe and scythe? And is the  
2 比如斧头、锄头、洒水壶、镐和镰刀？还有

1 character's current position a suitable place to use such a tool?  
2 角色当前的位置适合使用这种工具吗？

1 Then you should use use\_tool() instead of do\_action().  
2 那么你应该使用 use\_tool() 而不是 do\_action()。

1 5. Does the character already reach the target place?  
2 5. 角色已经到达目标地点了吗？

1 6. What was the previous action? If the previous action was a  
2 6. 上一个动作是什么？如果上一个动作是

1 movement, were you blocked?  
2 移动时，你被阻挡了吗？

1 7. If your task is to harvest the plant, did you water the seed?  
2 7. 如果你的任务是收获植物，你给种子浇水了吗？

1 The seeds only need to be watered ONCE every day. If you have  
2 种子每天只需浇水一次。如果你有

1 already watered the seed today, you should return home and go to  
2 今天已经给种子浇过水了，你应该回家去休息

1 sleep, waiting for the next day.  
2 睡觉，等待新的一天。

1 8. This is the most critical question. Based on the action rules  
2 8. 这是最关键的问题。基于行动规则

1 and self-reflection, what should be the most suitable action in  
2 以及自我反思，什么才是最合适行动

1 the valid action set for the next step? You should analyze the  
2 下一步的有效操作集是什么？你应该分析一下

1 effects of the action step by step. You should not repeat the  
2 逐步观察行动的效果。你不应重复

1 previous action again except for the movement action. Do not try  
2 除了移动动作外，之前的动作再次执行。不要尝试

1 to verify whether the previous action succeeded.  
2 验证先前操作是否成功。

1 9. Is the selected action the same as the last executed action? If  
2 9. 选定的动作是否与上次执行的动作相同？如果

1 yes, regenerate the action and give the reasons.  
2 是的，重新生成动作并给出理由。

1 10. Do all the selected actions exist in the valid action set? If  
2 10. 所有选定的动作都存在于有效动作集合中吗？如果

1 no, regenerate the action and give the reasons.  
2 不，重新生成动作并给出理由。

1 11. Analyze whether the selected action meets the requirements of  
2 11. 分析所选行动是否符合要求

1 the Actions below one by one. Does the generated action meet all  
2 下面逐一执行这些操作。生成的操作是否满足所有

1 the requirements? If not, regenerate the action and give the  
2 要求吗？如果没有，请重新生成操作并提供

1 reasons.  
2 原因。

Actions: The requirements that the generated action needs to follow.

操作：生成的操作需要遵循的要求。

1 The best action, or short sequence of actions without gaps, to  
2 最佳的动作，或连续无中断的短动作序列，以

1 execute next to progress in achieving the goal. Pay attention to  
2 执行下一步以推进目标的实现。注意

1 the names of the available skills and to the previous skills  
2 可用技能的名称及之前的技能

1 already executed, if any. You should also pay more attention to  
2 已执行的（如有）。您还应更加注意

1 the following action rules:  
2 以下操作规则：

1 1. You should output actions in Python code format and specify any  
2 1. 你应以Python代码格式输出操作，并说明任何

1 necessary parameters to execute that action. If the function has  
2 执行该操作所需的参数。如果函数具有

1 parameters, you should also include their names and decide their  
2 参数时，您还应包括它们的名称并确定它们的

1 values, like "move\_right (duration=1)". If it does not have a  
2 值，例如 "move\_right (duration=1)"。如果没有

1 parameter, just output the action, like "open\_map()".  
2 参数，只需输出动作，如"open\_map()"。

1 2. You can only output at most two actions in the output.  
2 2. 你最多只能输出两个操作。

1 3. In the screenshots, the blue band represents the left side and  
2 3. 在截图中，蓝色带表示左侧，且

1 the yellow band represents the right side. Please ignore character  
2 黄色带表示右侧。请忽略字符

1 's facing direction and output the action in an absolute direction  
2 面向方向并以绝对方向输出动作

1 like right and left.  
2 像左右一样。

1 4. If you want to interact with the objects in the toolbar, you  
2 4. 如果您想与工具栏中的对象进行交互，您

1 need to make sure that the target object is already selected. You  
2 需要确保目标对象已被选中。您

1 need to use select\_tool() to select them before executing use\_tool  
2 需要先使用select\_tool()选择它们，然后才能执行use\_tool

1 () or do\_action().  
2 () 或 do\_action()。

1 5. If you want to plant a seed or harvest a mature crop, please  
2 5. 如果您想种下种子或收获成熟的作物，请

1 use do\_action() instead of use\_tool(). If you want to use tool,  
2 使用 do\_action() 替代 use\_tool()。如果你想使用工具，

1 like axe, hoe, watering can, pickaxe and scythe, please use  
2 像斧头、锄头、水壶、镐和镰刀，请使用

1 use\_tool().  
2 use\_tool()。

1 6. If upon self-reflection you think the last action was  
2 6. 如果经过自我反思你认为上一个动作是

1 unavailable at the current place, you MUST move to another place.  
2 当前位置不可用，您必须移动到其他地方。

1 Please do not try to execute the same action again.  
2 请不要重复执行相同的操作。

1 7. If you want to get out of the house, just use the skill  
2 7. 如果你想出门，就用这个技能

1 get\_out\_of\_house(). You MUST NOT output any movement action behind  
2 离开房子()。你绝对不能在后面输出任何移动动作

1 this skill. And if the last executed action already contains this  
2 这项技能。如果最后执行的动作已经包含了这个

1 skill, do not execute this skill for the current step again.  
2 技能，不要在当前步骤再次执行此技能。

1 8. If upon self-reflection you think you were blocked, you MUST  
2 8. 如果经过自我反思你认为自己被阻止了，你必须

1 | change the direction of moving, so that you can pass obstacles.  
2 | 改变移动方向，以便你能通过障碍物。

1 | 9. You MUST NOT repeat the previous action again if you think the  
2 | 9. 如果你认为

1 | previous action fails.

2 | 先前的操作失败。

1 | 10. Your action should strictly follow the analysis in the

2 | 10. 你的操作应严格遵循分析结果

1 | reasoning. Do not output any additional action not mentioned in  
2 | 推理。不要输出未提及的任何额外操作

1 | the reasoning.

2 | 推理过程。

1 | 11. If you want to clear obstacles, you should follow the order of

2 | 11. 如果你想清除障碍，应按照以下顺序进行

1 | thinking as follows:

2 | 思考如下：

1 | - You MUST NOT move the character to the house.

2 | - 你绝对不能把角色移动到房子里。

1 | - In order to clear all obstacles below the house and make the

2 | - - 为了清除房屋下方的所有障碍物并使

1 | place suitable for cultivating, you should not target for a

2 | 适合耕作的地方，不应盲目追求

---

1 | specific obstacle. Instead, you should try your best to move the  
2 | 具体障碍。相反，你应该尽力移动

1 | character to pass every patch in the CLEARING AREA. You should  
2 | 角色需通过清理区域内的每个补丁。你应该

1 | clear every obstacle that blocks the character in this process.  
2 | 清除阻碍角色前进的所有障碍。

1 | - Every time after you move the character down (or up when  
2 | - 每次你将角色向下移动后 (或向上移动时

1 | being too far from the house), you should move the character right  
2 | (离房子太远时)，你应该向右移动角色

1 | or left (based on the character's position compared with the  
2 | 或向左 (基于角色与

1 | house) to fully explore the CLEARING AREA. You should clear all  
2 | 房屋) 以充分探索清理区域。你应该清理所有

1 obstacles the character meets in this process.  
2 角色在此过程中遇到的障碍。

1 - If you think the character has fully explored the current  
2 - - 如果你认为角色已经完全探索了当前

1 row of the CLEARING AREA, you should move the character down. If  
2 在清理区域 (CLEARING AREA) 的一行时, 你应该将角色向下移动。如果

1 there is an obstacle beneath the character, you should clear it  
2 角色下面有一个障碍物, 你应该清除它

1 first before moving the character down.  
2 先向前移动角色, 然后再向下移动。

1 - You should not move too far from the house. You should not  
2 - 你不应该离房子太远。你不应该

1 move the character down but should move him up instead if the  
2 向下移动角色, 但如果是, 则应向上移动他

1 house is not in the current screenshot.  
2 房屋不在当前截图中。

1 - You can take larger steps of moving left or right by  
2 - 你可以通过

1 adjusting the action's parameter. You MUST use a small parameter  
2 调整动作的参数。您必须使用一个较小的参数

1 when doing move\_down() to make sure the character only moves one  
2 在执行move\_down()时, 确保角色只移动一步

1 patch down.  
2 修补完成。

1 - If you think there is an obstacle in front of the character,  
2 - 如果你认为角色前方有障碍,

1 you should determine its type. You should then select the  
2 你应该确定它的类型。然后你应该选择

1 suitable tool by select\_tool() and clear the obstacle by use\_tool  
2 通过select\_tool()选择合适的工具并使用use\_tool清除障碍物

1 () .

1 - You should always use\_tool() after select\_tool(). Do not  
2 - 你应始终在 select\_tool() 之后使用 use\_tool()。不要

1 switch to another tool without using it.  
2 切换到另一个工具但不使用它。

1 - If the previous action contained use\_tool(), you MUST NOT  
2 - 如果前一个操作包含 use\_tool(), 则绝对不能

1 start with the same use\_tool() action in this round. (You can  
2 本轮继续使用相同的use\_tool()操作。 (你可以

1 still use use\_tool() by following a movement or select\_tool().)  
2 仍然通过跟随移动或选择工具(select\_tool())来使用use\_tool()。

1 - If the previous action contained use\_tool(), you should  
2 -- 如果上一个动作包含 use\_tool(), 你应当

1 determine whether the obstacle is cleared. If you are not sure  
2 判断障碍物是否已清除。如果不确定

1 that the obstacle is cleared, you are encouraged to try different  
2 障碍物是否清除，建议你尝试通过 select\_tool() 和 use\_tool()

1 tools by select\_tool() and use\_tool() before moving the character  
2 使用不同的工具，然后再移动角色

1 to other positions.  
2 到其他位置。

1 - If the previous action was a movement, you should determine  
2 -- 如果上一个动作是移动，你应当判断

1 whether there is an obstacle IN FRONT OF the character. If so, you  
2 角色前方是否有障碍物。如果有，

1 should select the suitable tool by select\_tool() and clear it by  
2 你应当通过 select\_tool() 选择合适的工具，并用

1 use\_tool().  
2 use\_tool() 清除它。

1 - If previous action contained use\_tool(), you should move the  
2 -- 如果上一个动作包含 use\_tool(), 你应当将

1 character to the same direction as before to test if the blocking  
2 角色向之前相同的方向移动，以测试阻挡的

1 obstacle is cleared.  
2 障碍物是否已被清除。

1 - If the blocking obstacle is not cleared, you should select a  
2 -- 如果阻挡障碍物未被清除，你应当选择

1 different tool to clear it.  
2 不同的工具来清除它。

You should only respond in the format described below, and you should  
你应仅以以下描述的格式回复，且

1 not output comments or other information.  
2 不得输出评论或其他信息。

Reasoning :

推理：

1. ...

2. ...

3. ...

Actions:

操作:

```
1  ```python
2
3
4      action (args1=x,args2=y)
5      操作 (参数1=x, 参数2=y)
6
7
8  ccc
9  ccc
10
11
12  ---
13
14
15
16 Prompt 18: Stardew: Information Gathering Shopping prompt.
17 提示18: 星露谷物语 (Stardew) : 信息收集购物提示。
18
19
20  ---
21
22
23
24 Assume you are a helpful AI assistant integrated with 'Stardew Valley'
25 假设你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手
26
27
28     on the PC, equipped to handle a wide range of tasks in the game.
29     在电脑上, 配备了处理游戏中各种任务的能力。
30
31
32     Your advanced capabilities enable you to process and interpret
33     您的高级能力使您能够处理和解读
34
35
36     gameplay screenshots and other relevant information.
37     游戏截图及其他相关信息。
38
39
40 <\$image_introduction\$>
41
42
43
44 Task overview:
45 任务概述:
46
47
48  ---
49
```

```
50
51
52 ---  

53
54
55
56 <\$task_description\$>
57
58
59
60 Current subtask:  

61 当前子任务:  

62
63
64 <\$subtask_description\$>
65
66
67
68 Description: Please analyze and describe the screenshot image in a  

69 描述: 请分析并描述截图图像中的内容
70
71
72 grid-by-grid format from left to right and top to bottom and then  

73 从左到右、从上到下逐格排列, 然后
74
75
76 provide an overall image description. Pay attention to anything  

77 提供整体图像描述。注意任何细节
78
79
80 related to the current subtask. The image is divided into a  $5 \times 3$   

81 与当前子任务相关。图像被划分为一个 $5 \times 3$ 
82
83
84 grid, each cell having its own coordinates. For each grid cell,  

85 网格, 每个单元格都有自己的坐标。对于每个网格单元,
86
87
88 describe the contents in detail, focusing on any critical icons,  

89 详细描述内容, 重点关注任何关键图标,
90
91
92 or objects present in that particular segment. If there are  

93 或该特定区域内存在的物体。
94
95
96 specific features such as characters or text, mention these as  

97 如果有特定特征如角色或文字, 也请一并提及。
98
99
100 well. After completing the description for one cell, proceed to  

101 完成一个单元格的描述后, 继续下一个, 例如, "在网格(1,1)中, [描述]。在网格
102
103
104 the next, for example, 'In grid (1,1), [description]. In grid  

105 (2,1)中, [描述]'。如此类推, 直到覆盖整个图像。
```

```
106
107
108     (2,1), [description]. ' and so on until the entire image is covered
109
110
111
112 Date_time: The date and time information in the game are shown on the
113 日期时间：游戏中的日期和时间信息显示在截图的右上角，位于网格(5,1)中。
114
115
116     upper-right of the screenshot, in grid (5, 1). An example of the
117 日期和时间信息的示例是"周三 10日, 下午5:10"。
118
119
120     date and time information is "Wed 10, 5:10 pm".
121
122
123
124 Energy: The current energy remains for the character doing actions.
125 能量：角色执行动作时的当前能量值。
126
127
128     The energy bar is shown on the bottom-right of the screenshot, in
129 能量条显示在截图的右下角，位于网格<b0></b0>中。
130
131
132     grid $\left( {5,3} \right)$. The full energy is 270 . An example of the energy
133 满能量为270。能量信息的示例是"150/270"。
134
135
136     information is "150/270".
137
138
139
140 Weather: The current weather information in the game, the weather is
141 天气：游戏中的当前天气信息，天气类型包括"晴天"、"雨天"、"有风"、"下雪"、"暴风雨"、"节日",
142
143
144     one from "Sunny", "Rainy", "Windy", "Snowy", "Stormy", "Festival",
145
146
147
148     "Wedding", and "null". If none of them applies, only output "null
149 "婚礼", 和"无"。如果都不适用，只输出"无"
150
151
152 .
153
154
155
156 Dialog: If there are some dialogs shown in the screenshot, extract the
157 对话：如果截图中显示有对话，提取
158
159
160     text of the conversation, like "Shopkeeper: What do you want to
161 对话内容，如"店主：你想买什么？"
```

162  
163  
164     buy?", otherwise, only output "null".  
165     , 否则只输出"无"。  
166  
167  
168 Other: Other information that does not belong to the above categories.  
169 其他: 不属于上述类别的其他信息。  
170  
171  
172     If none of them applies, only output "null".  
173     如果都不适用, 只输出"无"。  
174  
175  
176 You should only respond in the format described below and not output  
177 你应仅以以下描述的格式回复, 不要输出  
178  
179  
180     comments or other information.  
181     评论或其他信息。  
182  
183  
184 Description:  
185 描述:  
186  
187  
188 In grid \$\left( \begin{array}{c} 1,1 \\ 2,1 \end{array} \right)\$ , ... In grid \$\left( \begin{array}{c} 2,1 \\ 5,3 \end{array} \right)\$ , ... In grid \$\left( \begin{array}{c} 3,1 \\ 5,3 \end{array} \right)\$ , ... In grid \$\left( \begin{array}{c} 1,1 \\ 2,1 \\ 5,3 \end{array} \right)\$ ,  
189 在网格\$\left( \begin{array}{c} 1,1 \end{array} \right)\$中, ... 在网格\$\left( \begin{array}{c} 2,1 \end{array} \right)\$中, ... 在网格\$\left( \begin{array}{c} 3,1 \end{array} \right)\$中, ... 在网格\$\left( \begin{array}{c} 5,3 \end{array} \right)\$中,  
190  
191  
192     ... Overall, the image shows...  
193     ...整体来看, 图像显示了...  
194  
195  
196 Date\_time:  
197 日期时间:  
198  
199  
200 Date and time information  
201 日期和时间信息  
202  
203  
204 Energy:  
205 能量:  
206  
207  
208 The number of energy remains showing in the energy bar  
209 能量条中显示的剩余能量数值  
210  
211  
212 Weather:  
213 天气:  
214  
215

216 Weather information  
217 天气信息  
218  
219  
220 Dialog :  
221 对话:  
222  
223  
224 Dialog text  
225 对话文本  
226  
227  
228 Other:  
229 其他:  
230  
231  
232 Other information is ...  
233 其他信息是.....  
234  
235  
236 ---  
237  
238  
239  
240 Prompt 19: Stardew: Self-Reflection Shopping prompt.  
241 提示19: 星露谷物语 (Stardew) : 自我反思购物提示。  
242  
243  
244 ---  
245  
246  
247  
248 Assume you are a helpful AI assistant integrated with 'Stardew Valley'  
249 假设你是集成在"星露谷物语" (Stardew Valley) 中的一个有用的AI助手  
250  
251  
252 on the PC, equipped to handle a wide range of tasks in the game.  
253 运行于PC平台，能够处理游戏中的各种任务。  
254  
255  
256 Your advanced capabilities enable you to process and interpret  
257 你具备先进的能力，可以处理和解读  
258  
259  
260 gameplay screenshots and other relevant information. Your task is  
261 游戏截图及其他相关信息。你的任务是  
262  
263  
264 to examine these inputs, interpret the in-game context, and  
265 审查这些输入，解读游戏内的上下文，  
266  
267  
268 determine whether the executed action takes effect.  
269 并判断所执行的动作是否生效。  
270  
271

```
272 Target task:  
273 目标任务：  
274  
275  
276 <\$task_description\$>  
277  
278  
279  
280 Current subtask for completing the target task:  
281 完成目标任务的当前子任务：  
282  
283  
284 <\$subtask_description\$>  
285  
286  
287  
288 The reasoning for proposing the current subtask:  
289 提出当前子任务的理由：  
290  
291  
292 <\$subtask_reasoning\$>  
293  
294  
295  
296 Last executed action for completing the subtask:  
297 完成子任务的最后执行动作：  
298  
299  
300 <\$previous_action\$>  
301  
302  
303  
304 Reasoning for the last action:  
305 上一个动作的推理：  
306  
307  
308 <\$previous_reasoning\$>  
309  
310  
311  
312 Current Image description:  
313 当前图像描述：  
314  
315  
316 <\$image_description\$>  
317  
318  
319  
320 Toolbar information  
321 工具栏信息  
322  
323  
324 <\$toolbar_information\$>  
325  
326  
327
```

328 Summarization of recent history:  
329 近期历史总结:  
330  
331  
332 <\\$history\_summary\\$>  
333  
334  
335  
336 <\\$image\_introduction\\$>  
337  
338  
339  
340 Reasoning: You need to answer the following questions step by step to  
341 推理: 你需要逐步回答以下问题以  
342  
343  
344 get some reasoning based on the last action **and** sequential frames  
345 基于上一个动作和连续帧对角色进行推理  
346  
347  
348 of the character during the execution of the last action.  
349 在执行上一个动作期间的表现。  
350  
351  
352 1. Are the characters' positions in these frames identical?  
353 1. 这些帧中角色的位置是否相同?  
354  
355  
356 2. What **is** the executed action? Please answer this question **not** based  
357 2. 执行的动作是什么? 请不要基于  
358  
359  
360 on the sequential frames.  
361 连续帧来回答此问题。  
362  
363  
364 3. Was the executed action successful? Give reasons. You should refer  
365 3. 执行的动作是否成功? 请给出理由。你应参考  
366  
367  
368 to the following rules:  
369 以下规则:  
370  
371 - Analyze by observing given sequential frames **for** detailed  
372 - 通过观察给定的连续帧进行详细分析。  
373  
374  
375  
376 information.  
377 信息。  
378  
379  
380 - If the action involves moving forward, it **is** considered unsuccessful  
381 - 如果动作涉及向前移动, 只有当角色位置在  
382  
383

384       only when the character's position remains unchanged across  
385       连续帧中保持不变时，才视为不成功。  
386  
387  
388       sequential frames, regardless of background elements **and** other  
389       连续帧，无论背景元素和其他  
390  
391  
392       **people.**  
393       人物。  
394  
395  
396       - If you are **not** \${100}\%\$ sure that the action fails, regard it **as** success  
397       - 如果你不\${100}\%\$确定动作失败，则视为成功  
398  
399  
400       4. If the last action **is not** executed successfully, what **is** the most  
401       4. 如果最后一个动作未成功执行，最可能的原因是什么？  
402  
403  
404       probable cause? You should give only one cause **and** refer to the  
405       你应只给出一个原因，并遵循以下规则：  
406  
407  
408       following rules:  
409  
410  
411  
412       - The reasoning **for** the last action could be wrong.  
413       - 最后一个动作的推理可能是错误的。  
414  
415  
416       - If it **is** an interaction action such **as** buy\_item **or** do\_action, the  
417       - 如果是交互动作，如buy\_item（购买物品）或do\_action（执行动作），  
418  
419  
420       most probable cause was that the action was unavailable at the  
421       最可能的原因是该动作在当前位置不可用，  
422  
423  
424       current place, then you should move to a new place.  
425       那么你应移动到新地点。  
426  
427  
428       - If it **is** a movement action, the most probable cause was that you  
429       - 如果是移动动作，最可能的原因是你被可见或不可见的障碍物阻挡。  
430  
431  
432       were blocked by seen **or** unseen obstacles.  
433  
434  
435  
436       - If there **is** an error report, analyze the cause based on the report.  
437       - 如果有错误报告，应根据报告分析原因。  
438  
439

440 5. If the current subtask involves determining whether to enter the  
441 5. 如果当前子任务涉及判断是否进入商店，  
442  
443  
444 store, you need to compare the scene **in** the current screenshot  
445 你需要将当前截图中的场景与记忆中的截图场景进行比较，  
446  
447  
448 with the scene **in** the screenshot **from** Memory to determine whether  
449 以确定是否  
450  
451  
452 the character has entered the store, **if not**, then the task of  
453 角色已进入商店，如果没有，则任务为  
454  
455  
456 entering the store **is not** complete.  
457 进入商店尚未完成。  
458  
459  
460 6. Is the subtask completed? Give your reasons. If you want to make  
461 6. 子任务完成了吗？请说明理由。如果你想要继续  
462  
463  
464 any confirmation, regard it **as** a success. You should observe given  
465 任何确认都视为成功。你应当遵守所给的  
466  
467  
468 sequential frames, do **not** rely on the text information.  
469 连续帧，不依赖文本信息。  
470  
471  
472 7. Is the target task completed? Give your reasons.  
473 7. 目标任务完成了吗？请说明理由。  
474  
475  
476 8. If the current subtask involves purchase something, you should  
477 8. 如果当前子任务涉及购买某物，你应当  
478  
479  
480 check the toolbar **or** purchase menu to see **if** the purchase was  
481 检查工具栏或购买菜单以确认购买是否成功  
482  
483  
484 successful. Do **not** overbuy **or** miss the purchase.  
485 成功。不要过度购买或错过购买时机。  
486  
487  
488 9. Do you think the subtask **is** reasonable? Give your reasons.  
489 9. 你认为这个子任务合理吗？请说明理由。  
490  
491  
492 You should only respond **in** the format **as** described below.  
493 您应仅按照以下描述的格式回复。  
494  
495

496 Reasoning:  
497 推理：  
498  
499  
500 1. ...  
501  
502  
503  
504 2. ...  
505  
506  
507  
508 3. ...  
509  
510  
511  
512 ...  
513  
514  
515  
516 ---  
517  
518  
519  
520 Prompt 20: Stardew: Task Inference Shopping prompt.  
521 提示20：星露谷物语 (Stardew)：任务推断购物提示。  
522  
523  
524 ---  
525  
526  
527  
528 Assume you are a helpful AI assistant integrated with 'Stardew Valley'  
529 假设你是一个集成在《星露谷物语》(Stardew Valley)中的智能助手  
530  
531  
532 on the PC, equipped to handle a wide range of tasks in the game.  
533 在电脑上，配备了能够处理游戏中各种任务的功能。  
534  
535  
536 You will also be given a summary of the history that happened  
537 你还将获得一份发生过的历史摘要  
538  
539  
540 before the last screenshot. You should assist in summarizing the  
541 在最后一个截图之前。你应协助总结  
542  
543  
544 events for future decision-making and also propose a new subtask,  
545 事件以便未来决策，并提出一个新的子任务，  
546  
547  
548 which is the most suitable subtask for the current situation,  
549 即针对当前情况最合适的子任务，  
550  
551

552 given the target task.  
553 基于目标任务。  
554  
555  
556 Here **is** some helpful information to **help** you do the summarization **and**  
557 以下是一些有助于你进行总结和  
558  
559  
560 propose the subtask.  
561 提出子任务的有用信息。  
562  
563  
564 Current task:  
565 当前任务：  
566  
567  
568 <\\$task\_description\\$>  
569  
570  
571  
572 Previous proposed subtask **for** the task:  
573 之前为该任务提出的子任务：  
574  
575  
576 <\\$subtask\_description\\$>  
577  
578  
579  
580 Previous reasoning **for** proposing the subtask:  
581 提出该子任务的先前推理：  
582  
583  
584 <\\$subtask\_reasoning\\$>  
585  
586  
587  
588 <\\$image\_introduction\\$>  
589  
590  
591  
592 Current Image description:  
593 当前图像描述：  
594  
595  
596 <\\$image\_description\\$>  
597  
598  
599  
600 Last executed action:  
601 最后执行的动作：  
602  
603  
604 <\\$previous\_action\\$>  
605  
606  
607

608 Decision-making reasoning **for** the last executed action:  
609 对最后执行动作的决策推理:  
610  
611  
612 <\\$previous\_reasoning\$>  
613  
614  
615  
616 Self-reflection **for** the last executed action:  
617 对最后执行动作的自我反思:  
618  
619  
620 <\\$self\_reflection\_reasoning\$>  
621  
622  
623  
624 The following **is** the summary of history that happened before the last  
625 以下是发生在最后一个  
626  
627  
628 screenshot:  
629 截图之前的历史总结:  
630  
631  
632 <\\$previous\_summarization\$>  
633  
634  
635  
636 History\_summary: Summarize what happened **in** the past experience,  
637 历史总结: 总结过去经历中发生的事情,  
638  
639  
640 especially the last step according to the decision-making  
641 尤其是根据决策的最后一步  
642  
643  
644 reasoning **and** self-reflection reasoning **for** the last executed  
645 对最后执行的推理和自我反思推理  
646  
647  
648 action. The summarization needs to be precise, concrete **and** highly  
649 行动。总结需要精准、具体且高度  
650  
651 related to the task **and** follow the rules below.  
652 与任务相关并遵守以下规则。  
653  
654  
655  
656 1. Summarize the tasks **from** the history **and** the current task. What **is**  
657 1. 总结历史任务和当前任务。内容是什么  
658  
659  
660 the current progress of the task? For example, to harvest a seed,  
661 任务的当前进展? 例如, 收获种子时,  
662  
663

664 you need to water the seed **for** 4 days. And you have already  
665 你需要给种子浇水4天。你已经浇了  
666  
667  
668 planted the seed **and** watered it **for** two days.  
669 种下种子并浇水两天。  
670  
671  
672 2. Record the successful actions **and** organize them into events day by  
673 2. 记录成功的操作并将其按天整理成事件  
674  
675  
676 day.  
677 天。  
678  
679  
680 3. Do **not** forget the information **and** key events **in** the previous days.  
681 3. 不要忘记前几天的信息和关键事件。  
682  
683  
684 Subtask\_reasoning: Decide whether the previous subtask **is** finished **and**  
685 子任务推理：判断前一个子任务是否已完成并  
686  
687  
688 whether it **is** necessary to propose a new subtask. The subtask  
689 是否有必要提出一个新的子任务。该子任务  
690  
691  
692 should be straightforward, contribute to the target task **and** be  
693 应该是直接明了的，有助于目标任务，并且  
694  
695  
696 most suitable **for** the current situation, which should be completed  
697 最适合当前情况，应当完成  
698  
699  
700 within a few actions. You should respond to me **with**:  
701 在几步操作内。你应该回复我：  
702  
703 ---  
704  
705  
706  
707  
708 ---  
709  
710  
711  
712 1. How to finish the target task? You should analyze it step by step.  
713 1. 如何完成目标任务？你应该逐步分析。  
714  
715  
716 2. What **is** the current progress of the target task according to the  
717 2. 根据当前情况，目标任务的进展如何  
718  
719

720 analysis in step 1? Please do not make any assumptions if they are  
721 步骤1中的分析？如果有，请不要做任何假设  
722  
723  
724 not mentioned in the above information. You should assume that  
725 未在上述信息中提及。您应假设  
726  
727  
728 you are doing the task from scratch.  
729 你正在从头开始完成这项任务。  
730  
731  
732 3. What is the previous subtask? Does the previous subtask finish? If  
733 3. 上一个子任务是什么？上一个子任务完成了吗？如果  
734  
735 so, give evidence that the task was completed. Or is it improper  
736 那么，提供任务已完成的证据。还是说这样做不合适？  
737  
738  
739  
740 4. for the current situation? Then select a new one, otherwise you  
741 当前情况？那么请选择一个新的，否则你  
742  
743  
744 should reuse the last subtask.  
745 应重用最后一个子任务。  
746  
747  
748 4. If you want to propose a new subtask, give reasons why it is more  
749 4. 如果您想提出一个新的子任务，请说明其更为合理的原因  
750  
751  
752 5. The proposed subtask needs to be precise and concrete within one  
753 5. 所提出的子任务需要在一个范围内明确且具体  
754  
755  
760 sentence. It should not be related to any skills.  
761 句子。它不应与任何技能相关。  
762  
763  
764 6. Do not mention any grid information in the subtask description.  
765 6. 不要在子任务描述中提及任何网格信息。  
766  
767  
768 7. If the character does not reach the target place, you should  
769 7. 如果角色未能到达目标地点，您应当  
770  
771  
772 propose a movement task to make him closer to the target.  
773 提出一个移动任务，使他更接近目标。  
774  
775

776 8. If you want to purchase items, then you should move up to stand **in**  
777 8. 如果你想购买物品，应该走到  
778  
779  
780 front of the shopkeeper's counter, move slightly to align with the  
781 店主柜台前，稍微移动以对齐  
782  
783  
784 green counter **and** buy items. After purchasing, you can move down  
785 绿色柜台，然后购买物品。购买后，你可以向下  
786  
787 to the exit **and** leave store.  
789 走到出口离开商店。  
790  
791  
792 9. If you want to leave town, you should move along gray cobblestone  
793 9. 如果你想离开小镇，应沿着商店和诊所左侧的灰色鹅卵石  
794  
795  
796 road to the left of the store **and** the clinic.  
797 路前进。  
798  
799  
800 You should only respond **in** the format described below, **and** you should  
801 你只应按照下面描述的格式回复，且  
802  
803  
804 not output comments **or** other information.  
805 不要输出评论或其他信息。  
806  
807  
808 History\_summary:  
809 历史摘要：  
810  
811  
812 The summary **is**...  
813 摘要是.....  
814  
815  
816 Subtask\_reasoning:  
817 子任务推理：  
818  
819  
820 1. ...  
821  
822  
823  
824 2. ...  
825  
826  
827  
828 ...  
829  
830  
831

832 Subtask:  
833 子任务：  
834  
835  
836 The current subtask is  
837 当前子任务是  
838  
839  
840 ----  
841  
842  
843  
844 Prompt 21: Stardew: Action Planning Shopping prompt.  
845 提示21：星露谷(Stardew)：行动规划购物提示。  
846  
847  
848 ----  
849  
850  
851  
852 You are a helpful AI assistant integrated with 'Stardew Valley' on the  
853 你是集成在PC版《星露谷物语》(Stardew Valley)中的一个有用的AI助手，  
854  
855  
856 PC, equipped to handle various tasks in the game. Your advanced  
857 能够处理游戏中的各种任务。你的高级  
858  
859  
860 capabilities enable you to process and interpret gameplay  
861 功能使您能够处理和解读游戏截图  
862  
863  
864 screenshots and other relevant information. By analyzing these  
865 及其他相关信息。通过分析这些  
866  
867  
868 inputs, you gain a comprehensive understanding of the current  
869 输入，您可以全面了解游戏中的当前  
870  
871  
872 context and situation within the game. Utilizing this insight, you  
873 背景和情境。利用这些洞察，您  
874  
875  
876 are tasked with identifying the most suitable in-game action to  
877 的任务是根据当前任务确定最合适的游戏内操作  
878  
879  
880 take next, given the current task. You control the game character  
881 。您控制游戏角色  
882  
883  
884 and can execute actions from the available action set. Upon  
885 , 并可以从可用动作集中执行操作。评估  
886  
887

888 evaluating the provided information, your role **is** to articulate  
889所提供的信息后, 您的角色是明确  
890  
891  
892 the precise action you would deploy, considering the game's  
893 您将采取的具体行动, 考虑游戏的  
894  
895  
896 present circumstances, **and** specify **any** necessary parameters for  
897 当前状况, 并指定执行该行动所需的任何参数  
898  
899  
900 implementing that action.  
901 。  
902  
903  
904 Here **is** some helpful information to **help** you make the decision.  
905 以下是一些有助于您做出决策的有用信息。  
906  
907  
908 Current subtask:  
909 当前子任务:  
910  
911  
912 <\\$subtask\_description\\$>  
913  
914  
915  
916 Image description:  
917 图像描述:  
918  
919  
920 <\\$image\_description\\$>  
921  
922  
923  
924 Last executed action:  
925 上一次执行的动作:  
926  
927  
928 <\\$previous\_action\\$>  
929  
930  
931  
932 Reasoning **for** the last action:  
933 上一次动作的理由:  
934  
935  
936 <\\$previous\_reasoning\\$>  
937  
938  
939  
940 Self-**reflection** **for** the last executed action:  
941 对上一次执行操作的自我反思:  
942  
943

```
944 ---  
945  
946  
947 ---  
948 ---  
949  
950  
951  
952 <\$previous_self_reflection_reasoning\$>  
953  
954  
955  
956 Summarization of recent history:  
957 近期历史总结:  
958  
959  
960 <\$history_summary\$>  
961  
962  
963  
964 Valid action set in Python format to select the next action:  
965 以Python格式表示的有效动作集，用于选择下一步动作:  
966  
967  
968 <\$skill_library\$>  
969  
970  
971  
972 Grid System Information:  
973 网格系统信息:  
974  
975  
976 1. Each grid has a coordinate  $\left( \{x,y\} \right)$ . A larger  $x$  means that the grid  
is  
977 1. 每个网格都有一个坐标 $\left( \{x,y\} \right)$ 。较大的 $x$ 意味着该网格是  
978  
979  
980 on the more eastern (right) side, and a larger  $y$  means that the  
981 在更东侧（右侧），较大的 $y$ 值意味着  
982  
983  
984 grid is on the more southern (down) side. For example, moving from  
985 网格位于较南侧（下方）。例如，从  
986  
987  
988 grid (1,3) to grid (1,1) requires move_up(duration=2) and moving  
989 从网格(1,3)移动到网格(1,1)需要执行向上移动(duration=2)并保持移动状态  
990  
991  
992 from grid (1,1) to grid (2,1) requires move_right(duration=1)  
993 从网格(1,1)移动到网格(2,1)需要向右移动（持续时间=1）  
994  
995  
996 2. The larger the difference between the coordinates of the two grids,  
997 2. 两个网格坐标之间的差异越大，  
998
```

999  
1000 the longer it takes to move. Moving **from** grid (2,5) to grid (2,3)  
1001 移动所需时间越长。从网格(2,5)移动到网格(2,3)  
1002  
1003  
1004 takes longer than moving **from** grid  $\left( \begin{smallmatrix} 2 \\ 3 \end{smallmatrix} \right)$  to grid  $\left( \begin{smallmatrix} 1 \\ 3 \end{smallmatrix} \right)$ .  
1005 比从网格 $\left( \begin{smallmatrix} 2 \\ 3 \end{smallmatrix} \right)$ 移动到网格 $\left( \begin{smallmatrix} 1 \\ 3 \end{smallmatrix} \right)$ 花费更长时间。  
1006  
1007  
1008 <\\$image\_introduction\\$>  
1009  
1010  
1011  
1012 Based on the above information, analyze the current situation **and**  
1013 基于以上信息，分析当前状况并  
1014  
1015  
1016 provide the reasoning **for** what you should do **for** the **next** step to  
1017 说明你下一步应该做什么的理由  
1018  
1019  
1020 complete the task. Then, you should output the exact action you  
1021 完成任务。然后，你应输出确切的操作  
1022  
1023  
1024 want to execute **in** the game. You should respond to me **with**:  
1025 想在游戏中执行。你应该回复我：  
1026  
1027 Reasoning: You should think step by step **and** provide detailed  
1028 推理：你应该逐步思考并提供详细的  
1029  
1030  
1031 reasoning to determine the **next** action executed on the current  
1032 推理以确定当前执行的下一步操作  
1033  
1034  
1035 state of the task. You need to answer the following questions step  
1036 任务状态。您需要逐步回答以下问题  
1037  
1038  
1039  
1040 by step. You cannot miss the last question:  
1041 逐步进行。你绝不能错过最后一个问题是：  
1042  
1043  
1044 1. Does the character already reach the target place? You must  
1045 1. 角色已经到达目标地点了吗？你必须  
1046  
1047  
1048 move close enough to the **object** to be **in** contact **with** it **in** order  
1049 靠近物体直到接触为止  
1050  
1051  
1052 to interact **with** it. Just **in** the same grid **with** the target **is not**  
1053 与其互动。仅仅与目标处于同一网格并不足够

1054  
1055  
1056 enough.  
1057 足够。  
1058  
1059  
1060 2. Make use of the above image description, grid system  
1061 2. 利用上述图像描述和网格系统  
1062  
1063  
1064 information and current screenshot. Analyze whether the character  
1065 信息和当前截图。分析该角色是否  
1066  
1067  
1068 has reached the target place. You must move close enough to the  
1069 已到达目标地点。你必须靠近该地点。  
1070  
1071  
1072 object to be in contact with it in order to interact with it. Just  
1073 必须与其接触才能与之互动。仅仅  
1074  
1075  
1076 in the same grid with the target is not enough.  
1077 仅与目标处于同一网格是不够的。  
1078  
1079  
1080 3. What was the previous action? If the previous action was a  
1081 3. 上一个动作是什么？如果上一个动作是  
1082  
1083  
1084 movement, were you blocked?  
1085 移动时，你被阻挡了吗？  
1086  
1087  
1088 4. This is the most critical question. Based on the action rules  
1089 4. 这是最关键的问题。基于行动规则  
1090  
1091  
1092 and self-reflection, what should be the most suitable action in  
1093 以及自我反思，什么才是最合适的行动  
1094  
1095  
1096 the valid action set for the next step? You should analyze the  
1097 下一步的有效操作集？你应该分析  
1098  
1099  
1100 effects of the action step by step. You should not repeat the  
1101 逐步观察该行动的效果。你不应重复  
1102  
1103  
1104 previous action again except for the movement action. Do not try  
1105 除了移动动作外，重复之前的动作。不要尝试  
1106  
1107  
1108 to verify whether the previous action succeeded.  
1109 验证先前操作是否成功。

1110  
1111  
1112 5. Is the selected action the same as the last executed action? If  
1113 5. 所选动作是否与上次执行的动作相同? 如果  
1114  
1115  
1116 yes, regenerate the action and give the reasons.  
1117 是的, 重新生成动作并给出理由。  
1118  
1119  
1120 6. Do all the selected actions exist in the valid action set? If  
1121 6. 所有选定的动作是否都存在于有效动作集合中? 如果  
1122  
1123  
1124 no, regenerate the action and give the reasons.  
1125 不, 重新生成动作并给出理由。  
1126  
1127  
1128 7. Where is the player's character? Notice that the player's  
1129 7. 玩家角色在哪里? 注意玩家的  
1130  
1131  
1132 character is a brown-haired man wearing a blue jacket.  
1133 该角色是一个棕发男子, 穿着蓝色夹克。  
1134  
1135  
1136 8. Does the selected action contribute to the current subtask?  
1137 8. 所选操作是否有助于当前子任务?  
1138  
1139  
1140 9. Analyze whether the selected action meets the requirements of  
1141 9. 分析所选动作是否符合要求  
1142  
1143  
1144 the Actions below one by one. Does the generated action meet all  
1145 下面逐一执行这些操作。生成的操作是否满足所有  
1146  
1147  
1148 the requirements? If not, regenerate the action and give the  
1149 要求吗? 如果没有, 请重新生成操作并提供  
1150  
1151  
1152 reasons.  
1153 原因。  
1154  
1155  
1156 Actions: The requirements that the generated action needs to follow.  
1157 操作: 生成的操作需要遵循的要求。  
1158  
1159  
1160 The best action, or short sequence of actions without gaps, to  
1161 最佳的动作, 或连续无间断的短动作序列, 以  
1162  
1163  
1164 execute next to progress in achieving the goal. Pay attention to  
1165 执行下一步以推进目标的实现。注意

1166  
1167  
1168 the names of the available skills **and** to the previous skills  
1169 可用技能的名称及之前的技能  
1170  
1171  
1172 already executed, **if any**. You should also pay more attention to  
1173 已执行的（如有）。您还应更加注意  
1174  
1175  
1176 the following action rules:  
1177 以下操作规则：  
1178  
1179  
1180 1. You should output actions **in** Python code format **and** specify **any**  
1181 1. 你应以Python代码格式输出操作，并说明任何  
1182  
1183  
1184 necessary parameters to execute that action. If the function has  
1185 执行该操作所需的参数。如果函数具有  
1186  
1187  
1188 parameters, you should also include their names **and** decide their  
1189 参数时，您还应包括它们的名称并确定它们的  
1190  
1191  
1192 values, like "`move_right(duration=1)`". If it does **not** have a  
1193 值，比如 "`move_right(duration=1)`"。如果它没有  
1194  
1195  
1196 parameter, just output the action, like "`open_map()`".  
1197 参数，只需输出动作，如"`open_map()`"。  
1198  
1199  
1200 2. You can only output at most two actions **in** the output.  
1201 2. 你最多只能输出两个操作。  
1202  
1203  
1204 3. In the screenshots, the blue band represents the left side **and**  
1205 3. 在截图中，蓝色带表示左侧，且  
1206  
1207  
1208 the yellow band represents the right side. Please ignore character  
1209 黄色带表示右侧。请忽略字符  
1210  
1211  
1212 's facing direction and output the action in an absolute direction  
1213 面向方向并以绝对方向输出动作  
1214  
1215  
1216 like right **and** left.  
1217 像左右一样。  
1218  
1219  
1220 4. If upon **self**-reflection you think the last action was  
1221 4. 如果经过自我反思你认为上一个动作是

1222  
1223  
1224 unavailable at the current place, you MUST move to another place.  
1225 当前位置不可用，您必须移动到其他地方。  
1226  
1227  
1228 Please do not try to execute the same action again.  
1229 请不要重复执行相同的操作。  
1230  
1231  
1232 5. If you want to get out of the house, just use the skill  
1233 5. 如果你想出门，只需使用该技能  
1234  
1235  
1236 go\_through\_door. You MUST NOT output any movement action behind  
1237 通过门。你绝对不能输出任何移动动作  
1238  
1239  
1240 this skill. And if the last executed action already contains this  
1241 这项技能。如果最后执行的动作已经包含了这个  
1242  
1243  
1244 skill, do not execute this skill for the current step again.  
1245 技能，不要在当前步骤再次执行此技能。  
1246  
1247  
1248 6. If upon self-reflection you think you were blocked, you MUST  
1249 6. 如果经过自我反思你认为自己被阻止了，你必须  
1250  
1251  
1252 change the direction of moving, so that you can pass obstacles.  
1253 改变移动方向，以便你能通过障碍物。  
1254  
1255  
1256 7. You MUST NOT repeat the previous action again if you think the  
1257 7. 如果你认为  
1258  
1259 previous action fails.  
1260 先前的操作失败。  
1261  
1262  
1263  
1264 8. Your action should be strictly follow the analyze in the  
1265 8. 你的行动应严格遵循分析中的内容  
1266  
1267  
1268 reasoning. Do not output any additional action not mentioned in  
1269 推理。不要输出未提及的任何额外操作  
1270  
1271  
1272 the reasoning.  
1273 推理过程。  
1274  
1275  
1276 9. If the current subtask includes purchasing items, here are some  
1277 9. 如果当前子任务包括购买物品，以下是一些

1278  
1279  
1280 useful tips for you:  
1281 对你有用的提示：  
1282  
1283  
1284 - Pierre's store is east of the character's house.  
1285 - - 皮埃尔的商店在角色房子东边。  
1286  
1287  
1288 - if you do not see the store, you can move for a longer time  
1289 - - 如果你没有看到商店，可以多走一会儿  
1290  
1291  
1292 each time, such move\_right (duration=5). You can also move more  
1293 每次，执行此向右移动（持续时间=5）。你也可以移动更多  
1294  
1295  
1296 distance to the left each time to get home faster.  
1297 每次向左移动距离，以更快回家。  
1298  
1299  
1300 - To successfully enable the purchase transaction, you should  
1301 - 要成功完成购买交易，您应当  
1302  
1303  
1304 stand directly in front of the green counter, which left to the  
1305 直接站在绿色柜台前，柜台左侧是  
1306  
1307  
1308 white counter with word 'for sale'.  
1309 带有"出售"字样的白色柜台。  
1310  
1311  
1312 - After aligning with green counter, you should purchase items.  
1313 - - 与绿色计数器对齐后，您应购买物品。  
1314  
1315  
1316 - It is not necessary to positioned very precisely. If you stand  
1317 - - 不必非常精确地定位。如果你站着  
1318  
1319  
1320 near the green counter, you can try to purchase items.  
1321 在绿色柜台附近，你可以尝试购买物品。  
1322  
1323  
1324 10. If the current subtask includes exiting town and returning  
1325 10. 如果当前子任务包括离开城镇并返回  
1326  
1327  
1328 home, here are some useful tips for you:  
1329 家，这里有一些实用的建议给你：  
1330  
1331  
1332 - Character house is west of Pierre's store.  
1333 - 角色之家在皮埃尔商店的西边。

1334  
1335  
1336 - There is a long distance from home to the store, so each  
1337 - 从家到商店的距离很远, 所以每次  
1338  
1339  
1340 movement should take a long duration, such as move\_left(duration  
1341 移动应持续较长时间, 例如 move\_left(duration  
1342  
1343  
1344 \$= 5)\$ .  
1345  
1346  
1347  
1348 - Don't stand in the grass, move up and away from the lawn.  
1349 - - 不要站在草地上, 往上方远离草坪移动。  
1350  
1351  
1352 - The exit to the town is on the west(left) of Pierre's store and  
1353 - - 通往镇子的出口在皮埃尔商店和诊所的西侧 (左边)  
1354  
1355  
1356 clinic. You should move left along the stone road, which has a  
1357 。你应该沿着石头路向左走, 路下有一排木栅栏。  
1358  
1359  
1360 wooden fence below it. If you gets stuck, move up slightly to get  
1361 如果卡住了, 稍微向上移动以越过障碍物。  
1362  
1363  
1364 over the obstacle.  
1365  
1366  
1367  
1368 11. If you want to enter a building, you should use  
1369 11. 如果你想进入建筑物, 应使用  
1370  
1371  
1372 go\_through\_door(door="xxx\_entrance"); If you want to leave a  
1373 go\_through\_door(door="xxx\_entrance"); 如果你想离开建筑物, 应使用  
1374 go\_through\_door(door="xxx\_exit").  
1375  
1376 building, you should use go\_through\_door(door="xxx\_exit").  
1377  
1378  
1379  
1380 - You can use go\_through\_door(door="store\_entrance") to enter the  
1381 - - 你可以使用 go\_through\_door(door="store\_entrance") 进入商店。  
1382  
1383  
1384 store.  
1385  
1386  
1387  
1388 - You can use go\_through\_door(door="store\_exit") to leave the

1389 - - 你可以使用 `go_through_door(door="store_exit")` 离开商店。  
1390  
1391  
1392 `store.`  
1393  
1394  
1395  
1396 - You can use `go_through_door(door="home_entrance")` to enter your  
1397 - - 你可以使用 `go_through_door(door="home_entrance")` 进入你的房子。  
1398  
1399  
1400 `house.`  
1401  
1402  
1403  
1404 - You can use `go_through_door(door="home_exit")` to leave your  
1405 - - 你可以使用 `go_through_door(door="home_exit")` 离开你的房子。  
1406  
1407  
1408 `house.`  
1409  
1410  
1411  
1412 12. If you want align **with** the target, you MUST move slightly.  
1413 12. 如果你想与目标对齐，必须稍微移动。  
1414  
1415  
1416 Each movement take only 0.1 seconds, such **as** `move_xxx` (duration  
1417 每个动作仅需0.1秒，例如`move_xxx` (持续时间  
1418  
1419  
1420 `$= {0.1})$ .`  
1421  
1422  
1423  
1424 You should only respond **in** the format described below, **and** you should  
1425 你应仅以以下描述的格式回复，且应  
1426  
1427  
1428 not output comments **or** other information.  
1429 不输出评论或其他信息。  
1430  
1431  
1432 Reasoning :  
1433 推理：  
1434  
1435  
1436 1. ...  
1437  
1438  
1439  
1440 2. ...  
1441  
1442  
1443  
1444 3. ...

```
1445  
1446  
1447  
1448 Actions:  
1449 操作:  
1450  
1451  
1452 cccpython  
1453 cccpython  
1454  
1455  
1456     action(args1=x,args2=y)  
1457     action(args1=x,args2=y)  
1458  
1459  
1460 ccc  
1461 ccc  
1462  
1463  
1464 ---  
1465  
1466  
1467  
1468     ### H.4 Prompts for Dealer's Life 2  
1469     ### H.4 《Dealer's Life 2》提示语  
1470  
1471  
1472 Prompt 22: Dealer's Life 2: Information Gathering prompt.  
1473 提示22: 《Dealer's Life 2》信息收集提示。  
1474  
1475  
1476 ---  
1477  
1478  
1479  
1480 Assume you are a helpful AI assistant integrated with "Dealer's Life  
1481 假设你是集成于PC版《Dealer's Life 2》中的智能助手，  
1482  
1483  
1484 2" on the PC, equipped to handle a wide range of tasks in the game  
1485 能够处理游戏中的各种任务，  
1486  
1487 . Your advanced capabilities enable you to process and interpret  
1488 你的高级功能使你能够处理和解读  
1489  
1490  
1491  
1492 gameplay screenshots and other relevant information.  
1493 游戏截图及其他相关信息。  
1494  
1495  
1496 <\$image_introduction\$>  
1497  
1498  
1499  
1500 Current task:
```

1501 当前任务：  
1502  
1503  
1504 <\\$task\_description\\$>  
1505  
1506  
1507  
1508 Description: Please analyze **and** describe the screenshot image **in**  
1509 描述：请分析并描述截图图像中的内容  
1510  
1511  
1512 detail **and** then provide an overall image description. Most  
1513 详细描述，然后提供整体图像描述。最  
1514  
1515  
1516 importantly, identify the current page **type** **and** any relevant  
1517 重要的是，识别当前页面类型及与任务相关的  
1518  
1519 information related to the task. If there are specific features  
1520 信息。如果有特定特征  
1521  
1522  
1523 such **as** characters **or** text, mention these **as** well.  
1524 如人物或文本，也请提及。  
1525  
1526  
1527 Budget: Bank Balance **is** shown at the top right of the screenshot.  
1528 预算：银行余额显示在截图右上角。  
1529  
1530  
1531  
1532 Other: Other information that does **not** belong to the above categories.  
1533 其他：不属于上述类别的其他信息。  
1534  
1535  
1536 If none of them applies, only output "**null**".  
1537 如果都不适用，仅输出"null"。  
1538  
1539 You should only respond **in** the **format** described below **and** **not** output  
1540 你应仅以以下描述的格式回复，不要输出  
1541  
1542  
1543 comments **or** other information.  
1544 评论或其他信息。  
1545  
1546  
1547  
1548 Description:  
1549 描述：  
1550  
1551  
1552 The image shows...  
1553 图像显示...  
1554  
1555  
1556 Budget:

1557 预算：  
1558  
1559  
1560 The amount of budget  
1561 预算金额  
1562  
1563  
1564 Other:  
1565 其他：  
1566  
1567  
1568 Other information is ...  
1569 其他信息是...  
1570  
1571  
1572 ---  
1573  
1574  
1575  
1576 Prompt 23: Dealer's Life 2: Self Reflection prompt.  
1577 提示23: Dealer's Life 2: 自我反思提示。  
1578  
1579  
1580 ---  
1581  
1582  
1583  
1584 Assume you are a helpful AI assistant integrated with "Dealer's Life  
1585 假设你是一款集成在PC游戏"Dealer's Life 2"中的智能助手，  
1586  
1587  
1588 2" on the PC, equipped to handle a wide range of tasks in the game  
1589 能够处理游戏中的各种任务，  
1590  
1591  
1592 . Your advanced capabilities enable you to process and interpret  
1593 你具备先进的能力，可以处理和解析  
1594  
1595  
1596 gameplay screenshots and other relevant information. Your task is  
1597 游戏截图及其他相关信息。你的任务是  
1598  
1599  
1600 to examine these inputs, interpret the in-game context, and  
1601 审查这些输入，解读游戏内的情境，  
1602  
1603  
1604 determine whether the executed action takes effect.  
1605 并判断所执行的动作是否生效。  
1606  
1607  
1608 Target task:  
1609 目标任务：  
1610  
1611  
1612 <\\$task\_description\\$>

1613  
1614  
1615  
1616 Current subtask **for** completing the target task:  
1617 完成目标任务的当前子任务：  
1618  
1619  
1620 <\\$subtask\_description\\$>  
1621  
1622  
1623  
1624 The reasoning **for** proposing the current subtask:  
1625 提出当前子任务的理由：  
1626  
1627  
1628 <\\$subtask\_reasoning\\$>  
1629  
1630  
1631  
1632 Last executed action **for** completing the subtask:  
1633 完成子任务的上一次执行动作：  
1634  
1635  
1636 <\\$actions\\$>  
1637  
1638  
1639  
1640 Reasoning **for** the last action:  
1641 上一次动作的理由：  
1642  
1643  
1644 ---  
1645  
1646  
1647  
1648 ---  
1649  
1650  
1651  
1652 <\\$decision\_making\_reasoning\\$>  
1653  
1654  
1655  
1656 Current budget:  
1657 当前预算：  
1658  
1659  
1660 <\\$budget\\$>  
1661  
1662  
1663  
1664 Summarization of recent history:  
1665 近期历史总结：  
1666  
1667  
1668 <\\$history\_summary\\$>

1669  
1670  
1671  
1672 <\\$image\_introduction\\$>  
1673  
1674  
1675  
1676 Reasoning: You need to answer the following questions step by step to  
1677 推理：你需要逐步回答以下问题，  
1678  
1679  
1680 get some reasoning based on the last action **and** sequential frames  
1681 基于上一次动作及角色执行该动作时的连续帧，  
1682  
1683 of the character during the execution of the last action.  
1684 进行推理。  
1685  
1686  
1687  
1688 1. What **is** the executed action? Please answer this question **not** based  
1689 1. 执行的动作是什么？请不要基于连续帧来回答此问题  
1690  
1691  
1692 on the sequential frames.  
1693 。  
1694  
1695  
1696 2. Was the executed action successful? Give reasons. You should refer  
1697 2. 执行的动作是否成功？请给出理由。你应参考以下规则：  
1698  
1699  
1700 to the following rules:  
1701 。  
1702  
1703  
1704 - If you are **not** \${100}\%\$ sure that the action fails, regard it **as** success  
1705 - 如果你不\${100}\%\$确定动作失败，则视为成功  
1706  
1707  
1708 3. If the last action **is not** executed successfully, what **is** the most  
1709 3. 如果最后一个动作未成功执行，最可能的原因是什么？你应只给出一个原因并参考  
1710  
1711  
1712 probable cause? You should give only one cause **and** refer to the  
1713 以下规则：  
1714  
1715  
1716 following rules:  
1717 。  
1718  
1719  
1720 - The reasoning **for** the last action could be wrong.  
1721 - 最后一个动作的推理可能是错误的。  
1722  
1723  
1724 - If it **is** an interaction action, the most probable cause was that the

1725 - 如果是交互动作，最可能的原因是该动作在当前位置不可用，那么你应移动  
1726  
1727  
1728 action was unavailable at the current place, then you should move  
1729 到新的位置。  
1730  
1731  
1732 to a new place.  
1733 。  
1734  
1735  
1736 - If it **is** a movement action, the most probable cause was that you  
1737 - 如果是移动动作，最可能的原因是你被可见或不可见的障碍物阻挡。  
1738  
1739  
1740 were blocked by seen **or** unseen obstacles.  
1741 。  
1742  
1743  
1744 - If there **is** an error report, analyze the cause based on the report.  
1745 - 如果有错误报告，请根据报告分析原因。  
1746  
1747  
1748 4. Is the subtask completed? Give your reasons. If you want to make  
1749 4. 子任务完成了吗？请给出理由。如果你想  
1750  
1751  
1752 any confirmation, regard it **as** a success.  
1753 任何确认，都视为成功。  
1754  
1755  
1756 5. Is the target task completed? Give your reasons.  
1757 5. 目标任务完成了吗？请说明理由。  
1758  
1759  
1760 6. Do you think the subtask **is** reasonable? Give your reasons.  
1761 6. 你认为子任务合理吗？请说明理由。  
1762  
1763  
1764 Success: You need to output whether the last action was executed  
1765 成功：你需要输出最后一个动作是否执行  
1766  
1767  
1768 successfully **or** not.  
1769 成功。  
1770  
1771  
1772 - If the last action **is** successful, you should only output '**True**'.  
1773 - 如果最后一个动作成功，你只需输出"True"。  
1774  
1775  
1776 Otherwise, you should only output '**False**'.  
1777 否则，你只需输出"False"。  
1778  
1779  
1780 You should only respond **in** the **format** described below.

1781 你应仅以以下描述的格式回复。  
1782  
1783  
1784 Reasoning:  
1785 推理:  
1786  
1787  
1788 1. ...  
1789  
1790  
1791  
1792 2. ...  
1793  
1794  
1795  
1796 3. ...  
1797  
1798  
1799  
1800 Success:  
1801 成功:  
1802  
1803  
1804 True  
1805 True  
1806  
1807  
1808 ...  
1809  
1810  
1811  
1812 ----  
1813  
1814  
1815  
1816 Prompt 24: Dealer's Life 2: Task Inference prompt.  
1817 提示24: Dealer's Life 2: 任务推断提示。  
1818  
1819  
1820 ----  
1821  
1822  
1823  
1824 Assume you are a helpful AI assistant integrated with 'DealersLife2'  
1825 假设你是集成在PC端"DealersLife2"中的智能助手，  
1826  
1827  
1828 on the PC, equipped to handle a wide range of tasks in the game.  
1829 能够处理游戏中的各种任务。  
1830  
1831  
1832 You will also be given a summary of the history that happened  
1833 你还将获得最后一张截图之前发生的历史摘要，  
1834  
1835  
1836 before the last screenshot. You should assist in summarizing the

1837 你应协助总结这些内容。  
1838  
1839  
1840 events for future decision-making and also propose a new subtask,  
1841 为未来决策提供事件支持，并提出一个新的子任务，  
1842  
1843  
1844 which is the most suitable subtask for the current situation,  
1845 哪一个是当前情况最合适子任务，  
1846  
1847 given the target task.  
1848 鉴于目标任务。  
1849  
1850  
1851  
1852 Here is some helpful information to help you do the summarization and  
1853 这里有一些有助于你进行总结的有用信息  
1854  
1855  
1856 propose the subtask.  
1857 提出子任务。  
1858  
1859  
1860 Current task:  
1861 当前任务：  
1862  
1863  
1864 <\\\$task\_description\\\$>  
1865  
1866  
1867  
1868 Previous proposed subtask for the task:  
1869 之前为该任务提出的子任务：  
1870  
1871  
1872 ---  
1873  
1874  
1875  
1876 ---  
1877  
1878  
1879  
1880 <\\\$subtask\_description\\\$>  
1881  
1882  
1883  
1884 Previous reasoning for proposing the subtask:  
1885 提出子任务的先前推理：  
1886  
1887  
1888 <\\\$subtask\_reasoning\\\$>  
1889  
1890  
1891  
1892 <\\\$image\_introduction\\\$>

1893  
1894  
1895  
1896 Current budget:  
1897 当前预算:  
1898  
1899  
1900 <\\$budget\\$>  
1901  
1902  
1903  
1904 Current population:  
1905 当前人口:  
1906  
1907  
1908 <\\$population\\$>  
1909  
1910  
1911  
1912 Last executed action:  
1913 上次执行的操作:  
1914  
1915  
1916 <\\$actions\\$>  
1917  
1918  
1919  
1920 Decision-making reasoning for the last executed action:  
1921 最后执行动作的决策推理:  
1922  
1923  
1924 <\\$decision\_making\_reasoning\\$>  
1925  
1926  
1927  
1928 Self-reflection for the last executed action:  
1929 对上一次执行操作的自我反思:  
1930  
1931  
1932 <\\$self\_reflection\_reasoning\\$>  
1933  
1934  
1935  
1936 The following is the summary of history that happened before the last  
1937 以下是发生在上一次之前的历史总结  
1938  
1939  
1940 screenshot :  
1941 截图:  
1942  
1943  
1944 <\\$previous\_summarization\\$>  
1945  
1946  
1947  
1948 History\_summary: Summarize what happened in the past experience,

1949 历史总结：总结过去的经历发生了什么，  
1950  
1951  
1952 especially the last step according to the decision-making  
1953 特别是根据决策的最后一步  
1954  
1955  
1956 reasoning and self-reflection reasoning for the last executed  
1957 对最后执行的推理和自我反思推理  
1958  
1959  
1960 action. The summarization needs to be precise, concrete and highly  
1961 行动。总结需要精准、具体且高度  
1962  
1963  
1964 related to the task and follow the rules below.  
1965 与任务相关并遵守以下规则。  
1966  
1967  
1968 1. Summarize the tasks from the history and the current task. What is  
1969 1. 总结历史任务和当前任务。内容是什么  
1970  
1971  
1972 the current progress of the task?  
1973 任务的当前进展如何?  
1974  
1975  
1976 2. Record the successful actions and organize them into events day by  
1977 2. 记录成功的操作并将其按天整理成事件  
1978  
1979  
1980 day.  
1981 天。  
1982  
1983  
1984 3. Do not forget the information and key events in the previous days.  
1985 3. 不要忘记前几天的信息和关键事件。  
1986  
1987  
1988 4. If you are watering a seed. Record how many times you have watered  
1989 4. 如果你正在给种子浇水，记录浇水的次数  
1990  
1991  
1992 and calculate how many days you have to water before you can  
1993 并计算在你可以之前需要浇水多少天  
1994  
1995  
1996 harvest according to the toolbar information provided above.  
1997 根据上述工具栏信息进行收割。  
1998  
1999  
2000 Subtask\_reasoning: Decide whether the previous subtask is finished and  
2001 子任务推理：判断前一个子任务是否已完成并  
2002  
2003  
2004 whether it is necessary to propose a new subtask. The subtask

2005 是否有必要提出一个新的子任务。该子任务  
2006  
2007  
2008 should be straightforward, contribute to the target task **and** be  
2009 应该直截了当，有助于目标任务，并且是  
2010  
2011  
2012 most suitable **for** the current situation, which should be completed  
2013 最适合当前情况，应当完成  
2014  
2015  
2016 within a few actions. You should respond to me **with**:  
2017 在几步操作内。你应该回复我：  
2018  
2019  
2020 1. How to finish the target task? You should analyze it step by step.  
2021 1. 如何完成目标任务？你应该逐步分析。  
2022  
2023  
2024 2. What **is** the current progress of the target task according to the  
2025 2. 根据当前情况，目标任务的进展如何  
2026  
2027  
2028 analysis **in** step 1? Please do **not** make **any** assumptions **if** they are  
2029 步骤1中的分析？如果有，请不要做任何假设  
2030  
2031  
2032 not mentioned **in** the above information. You should assume that  
2033 上述信息中未提及。您应当假设  
2034  
2035  
2036 you are doing the task **from** scratch.  
2037 你正在从头开始完成这项任务。  
2038  
2039  
2040 3. What **is** the previous subtask? Does the previous subtask finish? Or  
2041 3. 上一个子任务是什么？上一个子任务完成了吗？还是  
2042  
2043  
2044 is it improper **for** the current situation? Then select a new one,  
2045 当前情况不合适吗？那就选择一个新的吧，  
2046  
2047  
2048 otherwise you should reuse the last subtask.  
2049 否则你应该重用最后一个子任务。  
2050  
2051  
2052 4. If you want to propose a new subtask, give reasons why it **is** more  
2053 4. 如果您想提出一个新的子任务，请说明其更为合理的原因  
2054  
2055  
2056 feasible **for** the current situation.  
2057 适合当前情况。  
2058  
2059  
2060 5. The proposed subtask needs to be precise **and** concrete within one

2061 5. 所提出的子任务需要在一个范围内明确且具体  
2062  
2063  
2064 sentence. It should **not** be related to **any** skills.  
2065 句子。它不应与任何技能相关。  
2066  
2067  
2068 6. Do **not** mention **any** grid information **in** the subtask description.  
2069 6. 不要在子任务描述中提及任何网格信息。  
2070  
2071  
2072 You should only respond **in** the **format** described below, **and** you should  
2073 您应仅以以下描述的格式回复，并且您应  
2074  
2075  
2076 **not** output comments **or** other information.  
2077 不输出评论或其他信息。  
2078  
2079  
2080 History\_summary :  
2081 历史概述：  
2082  
2083  
2084 The summary **is** ...  
2085 总结是.....  
2086  
2087  
2088 Subtask\_reasoning:  
2089 子任务推理：  
2090  
2091  
2092 1. ...  
2093  
2094  
2095  
2096 2. ...  
2097  
2098  
2099  
2100 3. ...  
2101  
2102  
2103  
2104 Subtask:  
2105 子任务：  
2106  
2107  
2108 The current subtask **is** ...  
2109 当前子任务是...  
2110  
2111  
2112 ---  
2113  
2114  
2115  
2116 Prompt 25: Dealer's Life 2: Action Planning prompt.

2117 提示25：发牌员生活2：行动计划提示。  
2118  
2119  
2120 ---  
2121  
2122  
2123  
2124 You are a helpful AI assistant integrated with "Dealer's Life 2" on  
2125 您是集成在"Dealer's Life 2"中的智能助理  
2126  
2127  
2128 the PC, equipped to handle various tasks in the game. Your  
2129 这台电脑配备了处理游戏中各种任务的能力。你的  
2130  
2131  
2132 advanced capabilities enable you to process and interpret gameplay  
2133 先进的功能使您能够处理和解析游戏玩法  
2134  
2135  
2136 screenshots and other relevant information. By analyzing these  
2137 截图及其他相关信息。通过分析这些  
2138  
2139  
2140 inputs, you gain a comprehensive understanding of the current  
2141 <inputs, you gain a comprehensive understanding of the current> 输入，您可以全面了解  
2142 当前情况  
2143  
2144 context and situation within the game. Utilizing this insight, you  
2145 <context>游戏中的背景和情境。利用这一洞察，您</context>  
2146  
2147  
2148 are tasked with identifying the most suitable in-game action to  
2149 负责识别最合适的游戏内动作以  
2150  
2151  
2152 take next, given the current task. You control the game character  
2153 根据当前任务，采取下一步行动。你控制游戏角色  
2154  
2155  
2156 and can execute actions from the available action set. Upon  
2157 并且可以从可用的动作集合中执行动作。接着  
2158  
2159  
2160 evaluating the provided information, your role is to articulate  
2161 评估所提供的信息，您的职责是阐述  
2162  
2163  
2164 the precise action you would deploy, considering the game's  
2165 考虑到游戏的当前情况，你将采取的具体行动  
2166  
2167  
2168 present circumstances, and specify any necessary parameters for  
2169 并指定执行该行动所需的任何参数。  
2170  
2171

2172                   implementing that action.  
2173                   实施该行动。  
2174  
2175  
2176 Here **is** some helpful information to **help** you make the decision.  
2177 以下是一些有助于你做出决策的信息。  
2178  
2179  
2180 Current subtask:  
2181 当前子任务：  
2182  
2183  
2184 <\\$subtask\_description\\$>  
2185  
2186  
2187  
2188 Current page **type**:  
2189 当前页面类型：  
2190  
2191  
2192 <\\$coordinates\\$>  
2193  
2194  
2195  
2196 Current budget:  
2197 当前预算：  
2198  
2199  
2200 <\\$budget\\$>  
2201  
2202  
2203  
2204 Last executed action:  
2205 上一次执行的动作：  
2206  
2207  
2208 <\\$actions\\$>  
2209  
2210  
2211  
2212 Reasoning **for** the last action:  
2213 上一次动作的推理：  
2214  
2215  
2216 <\\$decision\_making\_reasoning\\$>  
2217  
2218  
2219  
2220 Self-**reflection** **for** the last executed action:  
2221 对上一次执行动作的自我反思：  
2222  
2223  
2224 <\\$self\_reflection\_reasoning\\$>  
2225  
2226  
2227

2228 Summarization of recent history:  
2229 近期历史总结:  
2230  
2231  
2232 <\\$history\_summary\\$>  
2233  
2234  
2235  
2236 Valid action set in Python format to select the next action:  
2237 用于选择下一步动作的Python格式有效动作集:  
2238  
2239  
2240 <\\$skill\_library\\$>  
2241  
2242  
2243  
2244 <\\$image\_introduction\\$>  
2245  
2246  
2247  
2248 Based on the above information, analyze the current situation and  
2249 基于以上信息，分析当前情况并  
2250  
2251  
2252 provide the reasoning for what you should do for the next step to  
2253 提供你下一步应采取行动的理由，以完成任务。  
2254  
2255  
2256 complete the task. Then, you should output the exact action you  
2257 然后，你应输出你想在游戏中执行的具体动作。  
2258  
2259  
2260 want to execute in the game. You should respond to me with:  
2261 你应回复我：  
2262  
2263  
2264 Reasoning: You should think step by step and provide detailed  
2265 推理：你应该逐步思考并提供详细的  
2266  
2267 reasoning to determine the next action executed on the current  
2268 推理以确定当前执行的下一步操作  
2269  
2270  
2271 state of the task. You need to answer the following questions step  
2272 任务的状态。您需要逐步回答以下问题  
2273  
2274  
2275 by step. You cannot miss the last question:  
2276 逐步进行。你绝不能错过最后一个问  
2277 题：  
2278  
2279  
2280 1. Analyze the information in the screenshot. What can you observe  
2281 1. 分析截图中的信息。你能观察到什么?  
2282  
2283

2284                   in the screenshot? Please list some key elements.  
2285                   截图中有哪些关键元素？请列举一些。  
2286  
2287  
2288                   2. What is the current task? What are the requirements to achieve  
2289                   2. 当前任务是什么？实现该任务的要求有哪些?  
2290  
2291  
2292                   the goal?  
2293                   目标?  
2294  
2295  
2296                   3. What have you done so far in the game? What are the results of  
2297                   3. 你在游戏中到目前为止做了什么？结果如何  
2298  
2299  
2300                   the previous actions?  
2301                   之前的操作?  
2302  
2303  
2304                   ---  
2305  
2306  
2307  
2308                   ---  
2309  
2310  
2311  
2312                   4. What is your next step to achieve the goal? What is your plan?  
2313                   4. 你接下来的步骤是什么？你的计划是什么?  
2314  
2315  
2316                   Why do you choose this action? Please explain the reasoning behind  
2317                   你为什么选择这个动作？请说明背后的理由  
2318  
2319  
2320                   your decision.  
2321                   你的决定。  
2322  
2323  
2324                   5. If you were to respond to the customer's dialogue on the  
2325                   5. 如果你要回应客户的对话内容  
2326  
2327  
2328                   dialogue page, which of the listed responses in the screenshot  
2329                   对话页面，截图中列出的哪个回复  
2330  
2331  
2332                   would you choose? Why?  
2333                   你会选择什么？为什么？  
2334  
2335  
2336                   6. If you are to make an offer to a customer, how would you  
2337                   6. 如果你要向客户提出报价，你会如何做  
2338  
2339

2340 determine the price? You should determine the customer's role here  
2341 确定价格？你应该确定客户在这里的角色  
2342  
2343  
2344 . If the customer is a "seller", you should offer a price lower  
2345 如果客户是"卖家"，你应该提供一个更低的价格  
2346  
2347  
2348 than the item's value. If the customer is a "buyer", you should  
2349 比该商品的价值更高。如果客户是"买家"，你应该  
2350  
2351  
2352 offer a price higher than the item's value. Please explain your  
2353 出价高于物品价值。请说明您的理由  
2354  
2355  
2356 reasoning.  
2357 推理。  
2358  
2359  
2360 7. If the customer rejects your offer and makes a counteroffer,  
2361 7. 如果客户拒绝您的报价并提出还价，  
2362  
2363  
2364 what would you do? Would you accept the counteroffer or refuse the  
2365 你会怎么做？你会接受还价还是拒绝  
2366  
2367  
2368 deal? Why?  
2369 交易？为什么？  
2370  
2371  
2372 8. What does the current screen image show? Is it a giving price  
2373 8. 当前屏幕显示的是什么？是报价吗  
2374  
2375  
2376 page (it at least should show price \\$ in the right bottom of the  
2377 页面（至少应在右下角显示价格\\$）  
2378  
2379  
2380 screen image) or a non-giving price page and why?  
2381 屏幕图像）还是一个不显示价格的页面，为什么？  
2382  
2383  
2384 Actions: The requirements that the generated action needs to follow.  
2385 操作：生成的操作需要遵循的要求。  
2386  
2387  
2388 The best action, or short sequence of actions without gaps, to  
2389 最佳动作，或无间断的短动作序列，来  
2390  
2391  
2392 execute next to progress in achieving the goal. Pay attention to  
2393 执行下一步以推进目标的实现。注意  
2394  
2395

2396 the names of the available skills **and** the previous skills already  
2397 可用技能的名称以及之前已有的技能  
2398  
2399  
2400 executed, **if any**. You should also pay more attention to the  
2401 已执行的（如有）。你还应更加关注  
2402  
2403  
2404 following action rules:  
2405 以下操作规则：  
2406  
2407  
2408 1. You should output actions **in** Python code **format** **and** specify **any**  
2409 1. 你应以Python代码格式输出操作，并说明任何  
2410  
2411  
2412 necessary parameters to execute that action. If the function has  
2413 执行该操作所需的参数。如果函数具有  
2414  
2415  
2416 parameters, you should also include their names **and** decide their  
2417 参数时，您还应包括它们的名称并确定它们的  
2418  
2419  
2420 values, like "**move\_right(duration=1)**". If it does **not** have a  
2421 值，例如 "**move\_right(duration=1)**"。如果没有  
2422  
2423  
2424 parameter, just output the action, like "**open\_map()**".  
2425 参数，只需输出动作，如"**open\_map()**"。  
2426  
2427  
2428 2. Given the current situation **and** task, you should only choose  
2429 鉴于当前情况和任务，你应仅选择  
2430  
2431  
2432 the most suitable action **from** the valid action **set**. You cannot use  
2433 从有效动作集中选择最合适的作品。你不能使用  
2434  
2435  
2436 actions that are **not in** the valid action **set** to control the  
2437 不在有效动作集合中的动作用于控制  
2438  
2439  
2440 character.  
2441 字符。  
2442  
2443  
2444 3. In the screenshots, the blue band represents the left side **and**  
2445 3. 在截图中，蓝色带表示左侧，且  
2446  
2447  
2448 the yellow band represents the right side. Please ignore the  
2449 黄色带表示右侧。请忽略  
2450  
2451

2452 character's facing direction and output the action in an absolute  
2453 角色的朝向方向并以绝对方式输出动作  
2454  
2455  
2456 direction like right **and** left.  
2457 方向如右和左。  
2458  
2459  
2460 4. If you want to run **as** a successful dealer **in** conversation with  
2461 4. 如果你想在对话中成为一名成功的交易员  
2462  
2463  
2464 the customer, you should follow these rules:  
2465 对于客户，您应遵循以下规则：  
2466  
2467  
2468 4.1 Check the customer's dialogue.  
2469 4.1 检查客户的对话内容。  
2470  
2471  
2472 - If the customer **is** introducing himself **and** his purpose of  
2473 - 如果客户正在介绍自己及其来访您店铺的目的，  
2474  
2475  
2476 visiting your shop, you should always respond **with** "Let's see" to  
2477 您应始终以"我们看看"为回应，  
2478  
2479  
2480 make them potential buyers. This will be the first option **in** the  
2481 以将其转化为潜在买家。这将是对话中的第一个选项，您应选择它。  
2482  
2483  
2484 dialogue **and** you should select it.  
2485  
2486  
2487  
2488 4.2 Check the customer's response.  
2489 4.2 检查客户的回应。  
2490  
2491  
2492 - If the customer has shown you the details of the items **and** you  
2493 - 如果客户已向您展示了商品详情，且您已通过关闭商品详情页完成操作，  
2494  
2495  
2496 have completed by closing the item detail page, you should  
2497 您应以"我们成交"为回应以提出报价。  
2498  
2499  
2500 respond **with** "Let's deal" to make an offer. This will be the first  
2501 这将是对话中的第一个选项，您应选择它。  
2502  
2503  
2504 option **in** the dialogue **and** you should select it.  
2505  
2506  
2507

2508           5. If you want to run **as** a successful dealer **in** making an offer  
2509           5. 如果您想作为成功的交易商进行报价并决定是否接受报价或还价,  
2510  
2511  
2512           **and** deciding whether to take the offer **or** counteroffer, you should  
2513           您应遵循以下规则:  
2514  
2515  
2516           follow these rules:  
2517  
2518  
2519  
2520           5.1 Check the customer's role.  
2521           5.1 检查客户的角色。  
2522  
2523  
2524           - If the customer **is** a "**seller**", you should offer a price lower  
2525           - 如果客户是"卖家", 您应提出较低的价格  
2526  
2527  
2528           than the item's value. You should also consider your budget.  
2529           比物品的价值更高。你还应考虑你的预算。  
2530  
2531  
2532           - If the customer **is** a "**buyer**", you should offer a price higher  
2533           - 如果客户是"买家", 你应出价高于  
2534  
2535  
2536           than the item's value.  
2537           物品的价值。  
2538  
2539  
2540           5.2 Check the item's details.  
2541           5.2 检查物品的详细信息。  
2542  
2543  
2544           - You should check the item's "**rarity**", "**condition**", and "  
2545           - 你应检查物品的"稀有度"、"状况"和"  
2546  
2547  
2548           **estimate**" to determine the price you offer.  
2549           估价"以确定你出价的价格。  
2550  
2551  
2552           6. If you have opened up the buyer's or seller's character trait  
2553           6. 如果你已打开买家或卖家的性格特征  
2554  
2555  
2556           page, you should call the function to close the description page  
2557           页面, 应调用函数关闭描述页面  
2558  
2559  
2560           to proceed **with** the **next** action. You should NOT call **any** other  
2561           以继续下一步操作。你不应调用任何其他  
2562  
2563

2564 skill like dialogue ().  
2565 技能，如对话 ()。  
2566  
2567  
2568 7. Your action should strictly follow the analysis **in** the  
2569 7. 你的操作应严格遵循推理中的分析。  
2570  
2571  
2572 reasoning. Do **not** output **any** additional action **not** mentioned **in**  
2573 不要输出推理中未提及的任何额外操作。  
2574  
2575  
2576 the reasoning.  
2577  
2578  
2579  
2580 You should only respond **in** the **format** described below, **and** you should  
2581 你应仅以以下描述的格式回应，且不应  
2582  
2583  
2584 not output comments **or** other information.  
2585 输出评论或其他信息。  
2586  
2587  
2588 Reasoning:  
2589 推理:  
2590  
2591  
2592 1. ...  
2593  
2594  
2595  
2596 2. ...  
2597  
2598  
2599  
2600 3. ...  
2601  
2602  
2603  
2604 Actions:  
2605 操作:  
2606  
2607  
2608 cccpython  
2609 cccpython  
2610  
2611  
2612 action(args1=x,args2=y)  
2613 action(args1=x,args2=y)  
2614  
2615  
2616 ccc  
2617 ccc  
2618  
2619

2620 ---  
2621  
2622  
2623  
2624 **### H.5 Prompts for Software Applications**  
2625 **### H.5 软件应用提示**  
2626  
2627  
2628 Prompt 26: Chrome: Information Gathering prompt.  
2629 提示26: Chrome: 信息收集提示。  
2630  
2631  
2632 ---  
2633  
2634  
2635  
2636 Assume you are a helpful AI assistant integrated **with 'Google Chrome'**  
2637 假设你是集成在"Google Chrome" (谷歌浏览器) 中的智能助手  
2638  
2639     on the PC, equipped to handle a wide **range** of tasks **in the**  
2640     在PC上, 能够处理应用中的各种任务。  
2641  
2642  
2643  
2644     application. Your advanced capabilities enable you to process **and**  
2645     你具备先进的能力, 能够处理和  
2646  
2647  
2648     interpret application screenshots **and** other relevant information.  
2649     解读应用截图及其他相关信息。  
2650  
2651  
2652 Image introduction:  
2653 图片介绍:  
2654  
2655  
2656 <\\$image\_introduction\\$>  
2657  
2658  
2659  
2660 Overall task:  
2661 整体任务:  
2662  
2663  
2664 <\\$task\_description\\$>  
2665  
2666  
2667  
2668 Subtask description:  
2669 子任务描述:  
2670  
2671  
2672 <\\$subtask\_description\\$>  
2673  
2674  
2675

2676 Image\_Description:  
2677 图片描述:  
2678  
2679  
2680 1. Please describe the screenshot image **in** detail. Pay attention to  
2681 1. 请详细描述截图图像。注意  
2682  
2683  
2684 2. If the image includes a mouse cursor, please describe what UI  
2685 2. 如果图像中包含鼠标光标, 请描述鼠标当前位于哪个用户界面  
2686  
2687  
2688 3. Pay attention to **all** UI items **and** contents **in** the image. Do **not**  
2689 3. 注意图像中的所有用户界面项目和内容。不要对布局做出假设。  
2690  
2691  
2692 element the mouse **is** currently located near. Pay attention to the  
2693 元素附近。注意指针尖端的坐标, 而非鼠标光标的中心。  
2694  
2695  
2696 coordinates of the pointer tip, **not** the center of the mouse cursor  
2697 3. 注意图像中的所有用户界面项目和内容。不要对布局做出假设。  
2698  
2699  
2700  
2701  
2702  
2703  
2704  
2705  
2706  
2707  
2708 make assumptions about the layout.  
2709 不要对布局做出假设。  
2710  
2711  
2712 Description\_of\_bounding\_boxes:  
2713 边界框描述:  
2714  
2715  
2716 Please provide a **list** of EVERY bounding box **from** label ID of **1** to <  
2717 请逐一提供从标签ID **1**到 $\$length\_of\_som\_map\$$ 的每个边界框列表。标签ID标记在边界框的左  
上角。  
2718  
2719  
2720  
2721  
2722  
2723  
2724  
2725  
2726  
2727  
2728  
2729 upper left corner of the bounding boxes.  
标签ID标记在边界框的左上角。  
For bounding boxes containing text, provide ONLY the text.  
对于包含文本的边界框, 仅提供文本内容。

2730  
2731  
2732 For bounding boxes without text, brief description of the function.  
2733 对于不包含文本的边界框，简要描述其功能。  
2734  
2735  
2736 Format your response as follows: '1: function\_a', '2: text\_b', ...,  
2737 请按以下格式回复: '1: function\_a', '2: text\_b', ..., '\\$length\_of\_som\_map\$:  
function\_b'。不要写你不确定的内容。  
2738  
2739  
2740     \\$length\_of\_som\_map\$: function\_b'. Don't write anything you are  
2741     请按以下格式回复: '1: function\_a', '2: text\_b', ..., '\\$length\_of\_som\_map\$:  
2742     function\_b'。不要写你不确定的内容。  
2743  
2744     not sure about.  
2745     不要写你不确定的内容。  
2746  
2747  
2748 Target\_object\_name: Assume you can use an object detection model to  
2749 目标对象名称：假设你可以使用目标检测模型来  
2750  
2751  
2752     detect the most relevant object or UI item for completing the  
2753     检测完成当前任务所需的最相关对象或界面元素  
2754  
2755  
2756  
2757     current task if needed. What item should be detected to complete  
2758     如果需要。基于当前截图和当前任务，应检测哪个元素以完成  
2759  
2760  
2761     the task based on the current screenshot and the current task? You  
2762     该任务？你应遵守以下规则：  
2763  
2764     should obey the following rules:  
2765     你应遵守以下规则：  
2766  
2767  
2768 1. Identify an item that is relevant to the current or intermediate  
2769 1. 确定与当前或中间任务目标相关的元素。如果该元素位于截图中的边界框内，请包含对应的标签ID。  
2770  
2771  
2772     target of the task. If the item is within a bounding box in the  
2773     1. 确定与当前或中间任务目标相关的元素。如果该元素位于截图中的边界框内，请包含对应的标签  
2774 ID。  
2775  
2776     Screenshot, please include the corresponding label ID.  
2777     截图中，请包含对应的标签ID。  
2778  
2779  
2780 2. If no explicit item is specified, only output "null".  
2781 2. 如果没有明确指定元素，仅输出"null"。  
2782

2783  
2784 3. If there **is** no need to detect an **object**, only output "null".  
2785 3. 如果不需要检测对象，仅输出"null"。  
2786  
2787  
2788 Reasoning\_of\_object: Why was this **object** chosen, **or** why **is** there no  
2789 对象推理：为什么选择该对象，或者为什么不需要检测对象？  
2790  
2791  
2792     need to detect an **object**?  
2793     需要检测对象？  
2794  
2795  
2796 You should only respond **in** the **format** described below **and not** output  
2797 你应仅以以下描述的格式回复，不得输出评论或其他信息。不要更改每个  
2798  
2799  
2800 comments **or** other information. DO NOT change the title of each  
2801 项目的标题。  
2802  
2803  
2804     item.  
2805     项目的标题。  
2806  
2807  
2808 Image\_Description:  
2809 图像描述：  
2810  
2811  
2812 1. ...  
2813  
2814  
2815  
2816 2. ...  
2817  
2818  
2819  
2820 3. ...  
2821  
2822  
2823  
2824 Description\_of\_bounding\_boxes:  
2825 边界框描述：  
2826  
2827  
2828 Format like: 1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$  
2829 格式如：1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$  
2830  
2831  
2832     >: function\_b  
2833     >: function\_b  
2834  
2835  
2836 Target\_object\_name :  
2837 目标对象名称：  
2838

2839  
2840 label ID, Name  
2841 标签 ID, 名称  
2842  
2843  
2844 Reasoning\_of\_object:  
2845 对象推理:  
2846  
2847  
2848 ...  
2849  
2850  
2851  
2852 ---  
2853  
2854  
2855  
2856 Prompt 27: Chrome: Self-Reflection prompt.  
2857 提示 27: Chrome: 自我反思提示。  
2858  
2859  
2860 ---  
2861  
2862  
2863  
2864 Assume you are a helpful AI assistant integrated with 'Google Chrome,  
2865 假设你是集成在"Google Chrome"中的智能助手，  
2866  
2867 on the PC, equipped to handle a wide range of tasks in the  
2868 运行于PC端，具备处理应用程序中各种任务的能力。  
2869  
2870  
2871  
2872 application. Your advanced capabilities enable you to process and  
2873 你拥有先进的功能，能够处理和  
2874  
2875  
2876 interpret application screenshots and other relevant information.  
2877 解析应用程序截图及其他相关信息。  
2878  
2879  
2880 Your task is to examine these inputs, interpret the in-application  
2881 你的任务是审查这些输入，解读应用内  
2882  
2883  
2884 and OS context, and determine whether the executed action has  
2885 及操作系统上下文，并判断所执行的操作是否  
2886  
2887  
2888 taken the correct effect.  
2889 达到了预期效果。  
2890  
2891  
2892 Overall task description:  
2893 整体任务描述：  
2894

2895  
2896 <\\$task\_description\\$>  
2897  
2898  
2899  
2900 Image introduction:  
2901 图像介绍:  
2902  
2903  
2904 <\\$image\_introduction\\$>  
2905  
2906  
2907  
2908 Last executed action with parameters used:  
2909 最后执行的操作及所用参数:  
2910  
2911  
2912 <\\$previous\_action\_call\\$>  
2913  
2914  
2915  
2916 Implementation of the last executed action:  
2917 最后执行操作的实现:  
2918  
2919  
2920 <\\$action\_code\\$>  
2921  
2922  
2923  
2924 Error report for the last executed action:  
2925 最后执行操作的错误报告:  
2926  
2927  
2928 <\\$executing\_action\_error\\$>  
2929  
2930  
2931  
2932 Key reason for the last action:  
2933 最后操作的关键原因:  
2934  
2935  
2936 <\\$key\_reason\_of\_last\_action\\$>  
2937  
2938  
2939  
2940 History Summarization  
2941 历史总结  
2942  
2943  
2944 <\\$history\_summary\\$>  
2945  
2946  
2947  
2948 Success\_Detection flag for the overall task:  
2949 整体任务的成功检测 (Success\_Detection) 标志:  
2950

2951  
2952 <\\$success\_detection\\$>  
2953  
2954  
2955  
2956 Valid action **set** in Python **format** to select the **next** action:  
2957 用于选择下一操作的Python格式有效操作集:  
2958  
2959  
2960 <\\$skill\_library\\$>  
2961  
2962  
2963  
2964 Current **and** previous screenshot are the same:  
2965 当前截图与前一截图相同:  
2966  
2967  
2968 <\\$image\_same\_flag\\$>  
2969  
2970  
2971  
2972 Mouse position **in** the current screenshot **is** the same **as** **in** the  
2973 当前截图中的鼠标位置与  
2974  
2975  
2976 previous screenshot:  
2977 前一截图中的相同:  
2978  
2979  
2980 <\\$mouse\_position\_same\_flag\\$>  
2981  
2982  
2983  
2984 ---  
2985  
2986  
2987  
2988 ---  
2989  
2990  
2991  
2992 Self\_Reflection\_Reasoning:  
2993 自我反思推理:  
2994  
2995  
2996 You need to answer the following questions, step by step, to describe  
2997 你需要逐步回答以下问题，基于历史总结、最后操作及  
2998  
2999  
3000 your reasoning based on the history summarization, last action **and**  
3001 执行最后操作期间应用程序的连续截图，描述你的推理过程。  
3002  
3003  
3004 sequential screenshots of the application during the execution of  
3005  
3006

3007  
3008     the last action.  
3009  
3010  
3011  
3012 1. Please describe what the page **is** in the current screenshot. Respond  
3013 1. 请用一句话描述当前截图中的页面内容。  
3014  
3015  
3016     **in** one sentence.  
3017  
3018  
3019  
3020 2. What **is** the last executed action based on the text information  
3021 2. 根据文本信息，最后执行的操作是什么  
3022  
3023  
3024     **above?**  
3025     上方?  
3026  
3027  
3028 3. Was the last executed action successful? Give reasons. You should  
3029 3. 上一次执行的操作成功了吗？请说明理由。你应该  
3030  
3031  
3032     refer to the following rules:  
3033     请参照以下规则：  
3034  
3035  
3036 - If the last action executed was empty, then the previous action **is**  
3037 - 如果上一个执行的操作为空，则前一个操作是  
3038  
3039  
3040     deemed successful.  
3041     被视为成功。  
3042  
3043  
3044 - If the action involves moving the mouse, it **is** considered  
3045 - 如果操作涉及移动鼠标，则视为  
3046  
3047  
3048     unsuccessful when the mouse position remains unchanged **or** moves **in**  
3049     当鼠标位置保持不变或移动时，操作失败  
3050  
3051  
3052     an incorrect way across sequential screenshots, regardless of  
3053     在连续截图中错误的方式，无论如何  
3054  
3055  
3056     background elements **and** other items.  
3057     背景元素和其他项目。  
3058  
3059  
3060 - If the position to move the mouse to was incorrect **and** the mouse  
3061 - 如果鼠标要移动到的位置不正确，且鼠标  
3062

3063  
3064     didn't reach the target UI element, pay more attention to the  
3065     未达到目标用户界面元素，请更多关注  
3066  
3067  
3068     accurate coordinates to move to.  
3069     准确的移动坐标。  
3070  
3071  
3072     - If the operation involves **type** text, it will be considered  
3073     - 如果操作涉及文本类型，则将被考虑  
3074  
3075  
3076     unsuccessful when the corresponding text does **not** appear **in** the  
3077     当对应文本未出现时，操作失败  
3078  
3079  
3080     diagram, regardless of background elements **and** other items.  
3081     图表，无论背景元素和其他项目如何。  
3082  
3083  
3084     - If the action seemed to have no effect, pay attention to the latest  
3085     - 如果该操作似乎没有效果，请关注最新的  
3086  
3087  
3088     mouse position. Did it move? Did it get closer to the target UI  
3089     鼠标位置。它移动了吗？它是否更接近目标界面元素？  
3090  
3091  
3092     element? Where are the target coordinates **in** the action wrong? The  
3093     元素？动作中的目标坐标哪里出错了？  
3094  
3095  
3096     position of the mouse cursor on the screenshot shows their  
3097     截图上鼠标光标的位置显示了它们的  
3098  
3099  
3100     location.  
3101     位置。  
3102  
3103  
3104     - Was some unrelated UI item triggered by the last action?  
3105     - 上一个操作是否触发了某个无关的界面元素？  
3106  
3107  
3108     4. If the last action **is not** executed successfully, what **is** the most  
3109     4. 如果最后一个操作未成功执行，最重要的是  
3110  
3111  
3112     probable cause? You should give only one cause **and** refer to the  
3113     合理的理由？你只需给出一个原因并提及  
3114  
3115  
3116     following rules:  
3117     以下规则：  
3118

3119  
3120 - The reasoning **for** the last action could be wrong.  
3121 - 上一次操作的推理可能是错误的。  
3122  
3123  
3124 - If it was an action involving moving the mouse **or** the text cursor,  
3125 - 如果是涉及移动鼠标或文本光标的操作，  
3126  
3127  
3128     the most probable cause was that the coordinates used were  
3129     最可能的原因是所使用的坐标是  
3130  
3131  
3132     incorrect.  
3133     错误。  
3134  
3135  
3136 - If it **is** an interaction action, the most probable cause was that the  
3137 - 如果是交互动作，最可能的原因是  
3138  
3139  
3140     action was unavailable **or not** activated **in** the current state.  
3141     当前状态下操作不可用或未激活。  
3142  
3143  
3144 - If an unrelated change happened **in** the UI, the most probable cause  
3145 - 如果用户界面发生了无关的变化，最可能的原因是  
3146  
3147  
3148     was that the action triggered an incorrect UI element.  
3149     导致操作触发了错误的界面元素。  
3150  
3151  
3152 - If there **is** an error report, analyze the cause based on the report.  
3153 - 如果有错误报告，请根据报告分析原因。  
3154  
3155  
3156 Success\_Detection:  
3157 成功检测：  
3158  
3159  
3160 Based on the history summarization, the last action, the current  
3161 基于历史总结，最后的操作，当前的  
3162  
3163  
3164     screenshots **and** the Success\_Detection flag, determine whether the  
3165     截图和Success\_Detection标志，确定是否  
3166  
3167  
3168     overall task "<\\$task\_description\\$>" was successful. This  
3169     整体任务"<\\$task\_description\\$>"已成功完成。这  
3170  
3171  
3172     assessment should consider the overall task's success, not just  
3173     评估应考虑整体任务的成功，而不仅仅是  
3174

3175  
3176       individual actions.  
3177       个人行为。  
3178  
3179  
3180     - If the last action executed was an empty list and "<  
3181     - 如果最后执行的操作是一个空列表且"<  
3182  
3183  
3184       \\$success\_detection\\$" indicates the task is successful, then the  
3185       \\$success\_detection\\$" 表示任务成功, 然后  
3186  
3187  
3188       overall task has a high chance of being considered a success.  
3189       整体任务很可能被视为成功。  
3190  
3191  
3192     - If the overall task was unsuccessful, specify the reason of failure  
3193     - 如果整体任务未成功, 请说明失败原因  
3194  
3195  
3196       and which steps are missing.  
3197       以及缺少了哪些步骤。  
3198  
3199  
3200     - If the overall task was successful, ONLY output "SUCCESSFUL".  
3201     - 如果整体任务成功, 仅输出"SUCCESSFUL"。  
3202  
3203  
3204 You should only respond in the format as described below.  
3205 您应仅按照以下描述的格式回复。  
3206  
3207  
3208 Self\_Reflection\_Reasoning:  
3209 自我反思推理:  
3210  
3211 1. ...  
3212  
3213  
3214  
3215  
3216 2. ...  
3217  
3218  
3219  
3220 3. ...  
3221  
3222  
3223  
3224 Success\_Detection:  
3225 成功检测:  
3226  
3227  
3228 ...  
3229  
3230

3231  
3232 ---  
3233  
3234  
3235  
3236 ---  
3237  
3238  
3239  
3240 Assume you are a helpful AI assistant integrated with 'Google Chrome,  
3241 假设你是集成在"谷歌浏览器(Google Chrome)"中的一个有用的人工智能助手,  
3242  
3243  
3244 on the the PC, equipped to handle a wide range of tasks in the  
3245 在配备齐全的个人电脑(PC)上, 能够处理各种任务  
3246  
3247  
3248 game. You will be sequentially given <\\$event\_count\\$> screenshots  
3249 游戏。您将依次获得<\\$event\_count\\$>张截图  
3250  
3251  
3252 and corresponding descriptions of recent events. You will also be  
3253 以及近期事件的相应描述。您还将被  
3254  
3255  
3256 given a summary of the history that happened before the last  
3257 给出之前发生的历史摘要  
3258  
3259  
3260 screenshot. You should assist in summarizing the events for future  
3261 截图。您应协助总结事件以备将来参考  
3262  
3263  
3264 decision-making and also in proposing the most suitable subtask  
3265 决策制定以及提出最合适的子任务  
3266  
3267  
3268 to execute next, given the target task.  
3269 执行下一个, 针对目标任务。  
3270  
3271  
3272 Here is some helpful information to help you do the summarization and  
3273 这里有一些有助于你进行总结的有用信息  
3274  
3275  
3276 propose the subtask.  
3277 提出子任务。  
3278  
3279  
3280 Overall task description:  
3281 整体任务描述:  
3282  
3283  
3284 <\\$task\_description\\$>  
3285  
3286

3287  
3288 Previous proposed subtask **for** the task:  
3289 之前为该任务提出的子任务：  
3290  
3291  
3292 <\\$subtask\_description\\$>  
3293  
3294  
3295  
3296 Previous reasoning **for** proposing the subtask:  
3297 提出孩子任务的先前推理：  
3298  
3299  
3300 <\\$subtask\_reasoning\\$>  
3301  
3302  
3303  
3304 Image introduction:  
3305 图像介绍：  
3306  
3307  
3308 <\\$image\_introduction\\$>  
3309  
3310  
3311  
3312 Last executed action:  
3313 上次执行的操作：  
3314  
3315  
3316 <\\$previous\_action\\$>  
3317  
3318  
3319  
3320 Error report **for** the last executed action:  
3321 上次执行操作的错误报告：  
3322  
3323  
3324 <\\$executing\_action\_error\\$>  
3325  
3326  
3327  
3328 Key decision-making reasoning **for** the last executed action:  
3329 最后执行操作的关键决策推理：  
3330  
3331  
3332 <\\$previous\_reasoning\\$>  
3333  
3334  
3335  
3336 Self-reflection **for** the last executed action:  
3337 对上一次执行操作的自我反思：  
3338  
3339  
3340 <\\$self\_reflection\_reasoning\\$>  
3341  
3342

3343  
3344 Success\_Detection **for** the overall task:  
3345 整体任务的成功检测：  
3346  
3347  
3348 <\\$success\_detection\\$>  
3349  
3350  
3351  
3352 The following **is** the summary of history that happened before the last  
3353 以下是发生在上一次之前的历史概要  
3354  
3355  
3356 screenshot:  
3357 截图：  
3358  
3359  
3360 <\\$previous\_summarization\\$>  
3361  
3362  
3363  
3364 History\_summary: Summarize what happened **in** the past experience,  
3365 历史总结：总结过去的经历发生了什么，  
3366  
3367  
3368 especially the last step according to the decision-making  
3369 尤其是根据决策的最后一步  
3370  
3371  
3372 reasoning **and** self-reflection reasoning **for** the last executed  
3373 对最后执行的推理和自我反思推理  
3374  
3375  
3376 action. The summarization needs to be precise, concrete, highly  
3377 行动。总结需要精准、具体且高度  
3378  
3379  
3380 related to the task, **and** follow the rules below.  
3381 与任务相关，并遵守以下规则。  
3382  
3383  
3384 1. Determine **if** the task has been completed successfully. If it **is**  
3385 1. 确定任务是否已成功完成。如果是  
3386  
3387  
3388 successful, ignore question 2 to 5 .  
3389 成功，忽略问题2到5。  
3390  
3391  
3392 2. Summarize the tasks **from** the history **and** the current task. What  
3393 2. 总结历史任务和当前任务。什么  
3394  
3395  
3396 is the current progress of the task? For example, to open a file,  
3397 任务的当前进展如何？例如，打开一个文件时，  
3398

3399  
3400     you first need to select the file, then **open** it by clicking  
3401     你需要先选择文件，然后点击打开  
3402  
3403  
3404     somewhere **or** using the keyboard. Subtasks may have other pre-  
3405     在某处或使用键盘。子任务可能还有其他先决条件。  
3406  
3407  
3408     **requisites**.  
3409     先决条件。  
3410  
3411  
3412     3. Record the successful actions **and** organize them into events,  
3413     3. 记录成功的操作并将其组织成事件，  
3414  
3415  
3416     step by step.  
3417     逐步进行。  
3418  
3419  
3420     4. Which subtask has been completed? Which subtasks have **not**? Do  
3421     4. 哪个子任务已完成？哪些子任务尚未完成？不要忘记整体任务前面步骤中的信息和关键事件。  
3422  
3423  
3424     **not** forget the information **and** key events **in** the previous steps of  
3425     不要忘记整体任务前面步骤中的信息和关键事件。  
3426  
3427  
3428     the overall task.  
3429     整体任务。  
3430  
3431  
3432     Subtask\_reasoning: Decide whether the previous subtask **is** finished **and**  
3433     子任务推理：判断前一个子任务是否完成，是否有必要提出新的子任务。子任务应简明，能促进目标任务，  
并且最适合当前情况；应在几步操作内完成。你应回复以下内容。  
3434  
3435  
3436     whether it **is** necessary to propose a new subtask. The subtask  
3437     是否有必要提出新的子任务。子任务  
3438  
3439  
3440     ---  
3441  
3442  
3443  
3444     ---  
3445  
3446  
3447  
3448     should be straightforward, contribute to the target task, **and** be  
3449     应简明，能促进目标任务，并且  
3450  
3451  
3452     most suitable **for** the current situation; which should be completed  
3453     最适合当前情况；应在几步操作内完成。你应回复以下内容。

3454  
3455  
3456     within a few actions. You should respond **with** the following item.  
3457     你应回复以下内容。  
3458  
3459  
3460     1. Think about a hotkey related to the overall task **and next**  
3461     1. 思考一个与整体任务和下一个子任务相关的快捷键，请说明是什么。  
3462  
3463  
3464     subtask, please specify what it **is**.  
3465     2. 根据当前截图，找出完成任务的最直接且最简单的方法。  
3466  
3467  
3468     2. Based on the current screenshot, identify the most direct **and**  
3469     2. 根据当前截图，找出完成任务的最直接且最简单的方法。  
3470  
3471  
3472     easiest way to complete the task.  
3473     完成任务的最直接且最简单的方法。  
3474  
3475  
3476     3. Analyze the target task step by step to determine how to  
3477     3. 逐步分析目标任务以确定如何  
3478  
3479  
3480     complete it.  
3481     完成它。  
3482  
3483  
3484     4. What **is** the previous subtask? Has the previous subtask finished  
3485     4. 上一个子任务是什么？上一个子任务是否因自我反思而完成？  
3486  
3487  
3488     due to **self-reflection**? Or **is** it improper **for** the current  
3489     或者它是否不适合当前情况？  
3490  
3491  
3492     situation? If finished **or** improper, please select a new one,  
3493     如果已完成或不适合，请选择一个新的，  
3494  
3495  
3496     otherwise you should reuse the last subtask.  
3497     否则应重复使用上一个子任务。  
3498  
3499  
3500     5. If you want to propose a new subtask, give reasons why it **is**  
3501     5. 如果你想提出一个新的子任务，请说明为什么它  
3502  
3503  
3504     more feasible **for** the current situation. Please strictly follow  
3505     对当前情况更可行。请严格遵循  
3506  
3507  
3508     the description **and** requirements **in** the current task.  
3509     当前任务中的描述和要求。

3510  
3511  
3512     6. The proposed subtask needs to be precise **and** concrete within  
3513     6. 所提出的子任务需在一句话内准确具体,  
3514  
3515  
3516     one sentence. It should **not** be directly related to **any** skills.  
3517     且不应直接涉及任何技能。  
3518  
3519  
3520 You should only respond **in** the **format** described below, **and** you should  
3521 你应仅以以下描述的格式回复, 且不应  
3522  
3523  
3524     **not** output comments **or** other information.  
3525     输出评论或其他信息。  
3526  
3527  
3528 History\_summary:  
3529 历史摘要:  
3530  
3531  
3532     1. ...  
3533  
3534  
3535  
3536     2. ...  
3537  
3538  
3539  
3540     ...  
3541  
3542  
3543  
3544 Subtask\_reasoning:  
3545 子任务推理:  
3546  
3547  
3548     1. ...  
3549  
3550  
3551  
3552     2. ...  
3553  
3554  
3555  
3556     ...  
3557  
3558  
3559  
3560 Subtask\_description:  
3561 子任务描述:  
3562  
3563  
3564 The current subtask **is** ...  
3565 当前子任务是...

3566  
3567  
3568 ---  
3569  
3570  
3571  
3572 Prompt 29: Chrome: Action Planning prompt.  
3573 提示29: Chrome: 操作规划提示。  
3574  
3575  
3576 ---  
3577  
3578  
3579  
3580 You are a helpful AI assistant integrated with 'Google Chrome' on the  
3581 您是一款集成在"Google Chrome"（谷歌浏览器）上的智能助手，  
3582  
3583 PC, equipped to handle a wide range of tasks in the application.  
3585 运行于PC端，能够处理应用中的多种任务。  
3586  
3587  
3588 Your advanced capabilities enable you to process and interpret  
3589 您具备先进的能力，能够处理和解析  
3590  
3591  
3592 application screenshots and other relevant information. By  
3593 应用截图及其他相关信息。  
3594  
3595  
3596 analyzing these inputs, you gain a comprehensive understanding of  
3597 通过分析这些输入，您能够全面理解  
3598  
3599  
3600 the current context and situation within the application.  
3601 应用内当前的上下文和情境。  
3602  
3603  
3604 Utilizing these insights, you are tasked with identifying the most  
3605 利用这些洞察，您的任务是识别在当前  
3606  
3607  
3608 suitable in-application action to take next, given the current  
3609 任务下最合适的应用内下一步操作。  
3610  
3611  
3612 task. You control the application and can execute actions from the  
3613 您可以控制应用，并从可用操作集中执行操作以操控其界面。  
3614  
3615  
3616 available action set to manipulate its UI. Upon evaluating the  
3617 在评估所提供的信息后，您的职责是准确表达  
3618  
3619  
3620 provided information, your role is to articulate the precise  
3621 应执行的具体操作，考虑应用当前的

3622  
3623  
3624 actions you should perform, considering the application's present  
3625 状况，并指定实施该操作所需的任何参数。  
3626  
3627  
3628 circumstances, and specify any necessary parameters for  
3629  
3630  
3631  
3632 implementing that action.  
3633  
3634  
3635  
3636 Here is some helpful information to help you make the decision.  
3637 这里有一些有助于您做出决定的有用信息。  
3638  
3639  
3640 Overall task description:  
3641 整体任务描述:  
3642  
3643  
3644 <\\$task\_description\\$>  
3645  
3646  
3647  
3648 Subtask description:  
3649 子任务描述:  
3650  
3651  
3652 <\\$subtask\_description\\$>  
3653  
3654  
3655  
3656 Few shots:  
3657 少量示例:  
3658  
3659  
3660 <\\$few\_shots\\$>  
3661  
3662  
3663  
3664 Image introduction:  
3665 图像介绍:  
3666  
3667  
3668 <\\$image\_introduction\\$>  
3669  
3670  
3671  
3672 ---  
3673  
3674  
3675  
3676 ---  
3677

3678  
3679  
3680 Current **and** previous screenshot are the same:  
3681 当前截图与之前截图相同:  
3682  
3683  
3684 <\\$image\_same\_flag\\$>  
3685  
3686  
3687  
3688 Mouse position **in** the current screenshot **is** the same **as** **in** the  
3689 当前截图中的鼠标位置与  
3690  
3691  
3692 previous screenshot:  
3693 之前截图中的相同:  
3694  
3695  
3696 <\\$mouse\_position\_same\_flag\\$>  
3697  
3698  
3699  
3700 Description of current screenshot:  
3701 当前截图描述:  
3702  
3703  
3704 <\\$image\_description\\$>  
3705  
3706  
3707  
3708 Description of label IDs:  
3709 标签ID描述:  
3710  
3711  
3712 <\\$description\_of\_bounding\_boxes\\$>  
3713  
3714  
3715  
3716 Last executed action:  
3717 上一次执行的操作:  
3718  
3719  
3720 <\\$previous\_action\\$>  
3721  
3722  
3723  
3724 Key reason **for** the last action:  
3725 上一次操作的关键原因:  
3726  
3727  
3728 <\\$key\_reason\_of\_last\_action\\$>  
3729  
3730  
3731  
3732 Self-reflection **for** the last executed action:  
3733 对上一次执行操作的自我反思:

3734  
3735  
3736 <\\$previous\_self\_reflection\_reasoning\\$>  
3737  
3738  
3739  
3740 Summarization of recent history:  
3741 近期历史总结:  
3742  
3743  
3744 <\\$previous\_summarization\\$>  
3745  
3746  
3747  
3748 Valid action set in Python format to select the next action:  
3749 用于选择下一步操作的Python格式有效操作集:  
3750  
3751  
3752 <\\$skill\_library\\$>  
3753  
3754  
3755  
3756 Success detection for overall task:  
3757 整体任务的成功检测:  
3758  
3759  
3760 <\\$success\_detection\\$>  
3761  
3762  
3763  
3764 Based on the above information, you should first analyze the current  
3765 基于以上信息，您应首先分析当前  
3766  
3767 situation and provide the reasoning for what you should do for the  
3768 情况，并说明您应采取何种理由来完成  
3769  
3770  
3771  
3772 next step to complete the task. Then, you should output the exact  
3773 下一步任务。然后，您应输出在应用程序中  
3774  
3775  
3776 action you want to execute in the application.  
3777 想要执行的具体操作。  
3778  
3779  
3780 Pay attention to all UI items and contents in the image. DO NOT make  
3781 请注意图像中的所有界面元素和内容。不要对布局做出  
3782  
3783 assumptions about the layout! If the image includes a mouse cursor  
3784 假设！如果图像中包含鼠标光标，  
3785  
3786  
3787 , pay close attention to the coordinates of the pointer tip, not  
3788 请特别关注指针尖端的坐标，而非  
3789

3790  
3791  
3792     the centre of the mouse cursor.  
3793     鼠标光标的中心位置。  
3794  
3795  
3796 You should respond to me **with** the following information, **and** you MUST  
3797 您应按顺序向我反馈以下信息，且必须  
3798  
3799  
3800     respond one by one.  
3801     逐条回应。  
3802  
3803  
3804 Decision\_Making\_Reasoning: You should think step by step **and** provide  
3805 决策推理：您应逐步思考并提供  
3806  
3807     detailed reasoning to determine the **next** action executed on the  
3808     详细推理，以确定在当前任务状态下  
3809  
3810  
3811  
3812     current state of the task.  
3813     应执行的下一步操作。  
3814  
3815  
3816     1. Does "<\\$success\_detection\\$>" mean the overall task was  
3817     1."&lt;\$success\_detection\$&gt;"是否表示整体任务  
3818  
3819  
3820     successful? If successful, ignore questions 2 to 12.  
3821     成功？若成功，则忽略第2至12条问题。  
3822  
3823  
3824     2. Which skill **in** the Skill Library "<\\$skill\_library\\$>" has the  
3825     2."&lt;\$skill\_library\$&gt;"技能库中哪项技能具备  
3826  
3827  
3828     closest semantics to the current subtask "<\\$subtask\_description\\$  
3829     与当前子任务"<\\$subtask\_description\\$>"最接近的语义  
3830  
3831  
3832     >"? If there is an answer, select it as the output action.  
3833     >"? 如果有答案，请选择它作为输出操作。  
3834  
3835  
3836     3. Prefer keyboard operation instead of mouse operation. Are there  
3837     3. 优先使用键盘操作而非鼠标操作。是否有  
3838  
3839  
3840     any keyboard actions, such **as** using shortcut keys **or** pressing "  
3841     任何键盘操作，例如使用快捷键或按下"  
3842  
3843  
3844     enter", to finish the current step or overall task? If there is,  
3845     按"回车"键，完成当前步骤或整个任务？如果有，

3846  
3847  
3848 please specify which it **is**.  
3849 请具体说明是哪一种。  
3850  
3851  
3852 4. Based on the action rules, **self-reflection and previous**  
3853 4. 基于行动规则、自我反思和先前的  
3854  
3855  
3856 summarization, what should be the most suitable action **in** the  
3857 摘要, 最合适的操作应该是什么  
3858  
3859  
3860 valid action **set for the next step?** You should analyze the effects  
3861 下一步的有效操作集? 你应该分析其影响  
3862  
3863  
3864 of the action step by step.  
3865 逐步执行该操作。  
3866  
3867  
3868 5. If the previous action **is** unsuccessful, DO NOT repeat the  
3869 5. 如果前一步操作失败, 请勿重复执行  
3870  
3871  
3872 previous action, consider an alternative action **if** possible. If  
3873 先前的操作, 如有可能, 考虑替代操作。如果  
3874  
3875  
3876 there **is** an alternative action, please specify what it **is**, such **as**  
3877 如果有备选操作, 请具体说明, 例如  
3878  
3879  
3880 clicking different label IDs **or** using different shortcut keys.  
3881 点击不同的标签ID或使用不同的快捷键。  
3882  
3883  
3884 6. Always **try** pressing "**enter**" first instead of clicking it **with**  
3885 6. 总是先尝试按"回车"键, 而不是用鼠标点击它  
3886  
3887  
3888 the mouse, **if** the button you want to click **is** active.  
3889 鼠标, 如果你想点击的按钮是激活状态。  
3890  
3891  
3892 ---  
3893  
3894  
3895  
3896 ---  
3897  
3898  
3899  
3900 7. Check whether the UI element you want to operate exists **in** the  
3901 7. 检查您要操作的界面元素是否存在

3902  
3903  
3904 current screenshot. If **not**, you can choose to **return** to the  
3905 当前截图。如果不是，您可以选择返回  
3906  
3907  
3908 previous page **or** reopen a tab.  
3909 上一页或重新打开一个标签页。  
3910  
3911  
3912 8. In the current screenshot, identify the label ID of the  
3913 8. 在当前截图中，识别标签ID  
3914  
3915  
3916 bounding box most relevant to the current step. If there **is** text  
3917 与当前步骤最相关的边界框。如果有文本  
3918  
3919  
3920 within this bounding box, please provide the text.  
3921 请在此边界框内提供文本。  
3922  
3923  
3924 9. If mouse actions are necessary, use that specific bounding box  
3925 9. 如果需要鼠标操作，请使用该特定的边界框  
3926  
3927  
3928 label ID (**if shown in** the current screenshot) **as** a parameter,  
3929 将标签ID（如果在当前截图中显示）作为参数，  
3930  
3931  
3932 rather than directly generating normalized \$x\$ and \$y\$ coordinates. If  
3933 而不是直接生成归一化的\$x\$和\$y\$坐标。如果  
3934  
3935  
3936 there **is any** relevant label ID, please specify which it **is**.  
3937 如果有相关的标签ID，请指明是哪一个。  
3938  
3939  
3940 10. If a dialog box appears, make sure to check the content of the  
3941 10. 如果出现对话框，请务必检查内容  
3942  
3943  
3944 dialog box to determine **if** the task **is** complete. For instance,  
3945 用于确定任务是否完成的对话框。例如，  
3946  
3947  
3948 when a download dialog box appears, the task **is** only completed  
3949 当出现下载对话框时，任务才算完成  
3950  
3951  
3952 after pressing the Enter key **or** clicking "Save".  
3953 按下回车键或点击"保存"后。  
3954  
3955  
3956 11. If you need to use an action outside an **open** menu **or** dialog  
3957 11. 如果需要在打开的菜单或对话框之外使用某个操作

3958  
3959  
3960 box, please close the current menu or dialog box before trying the  
3961 对话框, 请先关闭当前菜单或对话框后再尝试  
3962  
3963  
3964 next action.  
3965 下一步操作。  
3966  
3967  
3968 12. If you anticipate that the next step involves typing text,  
3969 12. 如果你预期下一步需要输入文本,  
3970  
3971 confirm that the last executed action was a click at the  
3972 确认最后执行的操作是在  
3973  
3974  
3975 appropriate input box. If not, it is mandatory to click on the  
3976 适当的输入框。如果没有, 必须点击  
3977  
3978 corresponding input box before proceeding with typing.  
3979 在继续输入之前, 请先点击相应的输入框。  
3980  
3981  
3982 Actions: The best action, or short sequence of actions without gaps,  
3983 操作: 最佳操作, 或无间断的短操作序列,  
3984  
3985  
3986 to execute next to progress in achieving the goal. Pay attention  
3987 执行下一步以推进目标的实现。请注意  
3988  
3989  
3990  
3991 to the names of the available skills and the previous skills  
3992 到可用技能的名称和之前的技能  
3993  
3994  
3995 already executed, if any. Pay special attention to the coordinates  
3996 已执行 (如有)。特别注意坐标  
3997  
3998  
3999 of any action that needs them. Do not make assumptions about the  
4000 任何需要它们的行动。不要对其做出假设  
4001  
4002  
4003  
4004 location of UI elements or their coordinates, analyse in detail  
4005 用户界面元素的位置或其坐标, 进行详细分析  
4006  
4007  
4008 any provided images. You should also pay more attention to the  
4009 任何提供的图像。你还应更加关注  
4010  
4011  
4012 following action rules:  
4013 以下操作规则:

4014  
4015  
4016 1. If "<\\$success\_detection\\$>" means the overall task was  
4017 1. 如果"<\\$success\_detection\\$>"表示整体任务已完成  
4018  
4019  
4020 successful or equal to "True", then the output action MUST be  
4021 成功或等于"True", 则输出操作必须是  
4022  
4023  
4024 empty like ". Be careful to check the task was really successful.  
4025 像""一样空。请务必确认任务确实成功完成。  
4026  
4027  
4028 2. You should output actions in Python code format and specify any  
4029 2. 你应以Python代码格式输出操作，并指定执行该操作所需的参数。  
4030  
4031  
4032 necessary parameters to execute that action. Only use function  
4033 只使用有效操作集中的函数名和参数名。  
4034  
4035  
4036 names and argument names exactly as shown in the valid action set.  
4037 函数名和参数名必须与有效操作集中的完全一致。  
4038  
4039  
4040 If a function has parameters, you should also include their names  
4041 如果函数有参数，也应包括参数名并决定其值，如 "press\_shift(duration=1)"。  
4042  
4043  
4044 and decide their values, like "press\_shift (duration=1)". If it  
4045 如果没有参数，则只需输出操作，如 "release\_mouse\_buttons() "。  
4046  
4047  
4048 does not have a parameter, just output the action, like "  
4049  
4050  
4051 release\_mouse\_buttons ()".  
4052  
4053  
4054  
4055  
4056  
4057 3. Before typing text, ensure that the last executed action  
4058 3. 在输入文本前，确保上一个执行的操作是点击了相关输入框。  
4059  
4060 involved clicking on the relevant input box. If the last action  
4061 如果上一个操作不是点击该输入框，则必须先点击对应输入框再继续。  
4062  
4063  
4064 was not a click on this input box, the required action MUST be to  
4065  
4066  
4067  
4068 click on the corresponding input box before proceeding.  
4069

4070  
4071  
4072     4. Given the current situation **and** task, you should only choose  
4073         4. 根据当前情况和任务，只能从有效操作集中选择最合适的操作。  
4074  
4075  
4076         the most suitable action **from** the valid action **set**. You cannot use  
4077         不能使用不在有效操作集中的操作来控制应用程序。  
4078  
4079  
4080         actions that are **not in** the valid action **set** to control the  
4081  
4082  
4083  
4084         application.  
4085  
4086  
4087  
4088         5. When you perform a mouse action, always select the target UI  
4089         5. 执行鼠标操作时，始终选择目标UI元素。  
4090  
4091  
4092         element closest to the UI element of the previous action **for**  
4093         最接近上一个操作的UI元素的元素  
4094  
4095  
4096         operation.  
4097         操作。  
4098  
4099  
4100         6. When you decide to operate on a file, such **as** downloading it,  
4101         6. 当您决定对文件进行操作时，例如下载，  
4102  
4103  
4104         please pay attention to the path **and** name of the current file.  
4105         请注意当前文件的路径和名称。  
4106  
4107  
4108         Key\_reason\_of\_last\_action: Summarize the key reasons why you output  
4109         上一次操作的关键原因：总结您执行此操作的主要原因。  
4110  
4111  
4112         this action.  
4113         此操作。  
4114  
4115  
4116         You should only respond **in** the **format** described below. In your  
4117         您应仅以以下描述的格式进行回复。在您选择操作的  
4118  
4119  
4120         reasoning **for** the chosen actions, also describe which item you  
4121         推理中，还应描述您决定交互的项目及原因。不要更改每个  
4122  
4123  
4124         decided to interact **with** and why. DO NOT change the title of each  
4125         项目的标题。除以下格式外，不应输出其他评论或信息。

```
4126  
4127  
4128     item. You should not output other comments or information besides  
4129  
4130  
4131  
4132     the format below.  
4133  
4134  
4135  
4136 Decision_Making_Reasoning:  
4137     决策推理:  
4138  
4139  
4140     1. ...  
4141  
4142  
4143  
4144     2. ...  
4145  
4146  
4147  
4148     ---  
4149  
4150  
4151  
4152     3. ...  
4153  
4154  
4155  
4156     ---  
4157  
4158  
4159  
4160     ...  
4161  
4162  
4163  
4164     Actions:  
4165     操作:  
4166  
4167  
4168     44 python  
4169     44 python  
4170  
4171  
4172     action(args1=x,args2=y)  
4173     action(args1=x,args2=y)  
4174  
4175  
4176     ccc  
4177     ccc  
4178  
4179  
4180     Key_reason_of_last_action:  
4181     上次操作的主要原因:
```

4182  
4183  
4184 ...  
4185  
4186  
4187  
4188 ---  
4189  
4190  
4191  
4192 Prompt 30: Outlook: Information Gathering prompt.  
4193 提示30：展望：信息收集提示。  
4194  
4195  
4196 ---  
4197  
4198  
4199  
4200 You an expert helpful AI assistant which follows instructions **and**  
4201 你是一位遵循指令的专业且乐于助人的人工智能助手  
4202  
4203  
4204 performs desktop computer tasks **as** instructed. You have expert  
4205 按照指示执行桌面计算机任务。您具备专家级  
4206  
4207  
4208 knowledge of '**Microsoft Outlook**' on the PC **and** can handle a wide  
4209 具备PC端"Microsoft Outlook"（微软Outlook）的知识，能够熟练操作多种功能  
4210  
4211  
4212 range of tasks **in** the application using the keyboard, shortcut  
4213 使用键盘快捷键完成应用中的一系列任务  
4214  
4215  
4216 keys, **and** mouse operations. For each step, you will get one **or**  
4217 键盘和鼠标操作。每一步，您将获得一个或  
4218  
4219  
4220 more observation images, which are screenshots of the computer  
4221 更多观测图像，即计算机屏幕截图  
4222  
4223  
4224 screen. Your advanced capabilities enable you to process **and**  
4225 屏幕。您先进的能力使您能够处理和  
4226  
4227  
4228 interpret these application screenshots **and** other relevant  
4229 解释这些应用程序截图及其他相关内容  
4230  
4231  
4232 information **in** detail. The screenshots include numerical tags ( )  
4233 详细的信息。截图中包含数字标签 ( )  
4234  
4235  
4236 label IDs) **and** bounding boxes marking some UI items.  
4237 <label IDs>和标记某些用户界面(UI)项目的边界框。

4238  
4239  
4240 Image introduction:  
4241 图像介绍:  
4242  
4243  
4244 <\\$image\_introduction\\$>  
4245  
4246  
4247  
4248 Overall task:  
4249 整体任务:  
4250  
4251  
4252 <\\$task\_description\\$>  
4253  
4254  
4255  
4256 Subtask description:  
4257 子任务描述:  
4258  
4259  
4260 <\\$subtask\_description\\$>  
4261  
4262  
4263  
4264 Image\_Description:  
4265 图像描述:  
4266  
4267  
4268 1. Please describe the screenshot image **in** detail. Pay attention to  
4269 1. 请详细描述截图图像。请注意  
4270  
4271  
4272 any details **in** the image, **if any**, especially critical icons, open  
4273 图像中的任何细节，尤其是关键图标，均已打开  
4274  
4275  
4276 menus **or** dialogs, **and** any instructions **for** the application user.  
4277 菜单或对话框，以及应用程序用户的任何指示。  
4278  
4279  
4280 Focus on the image contents **and** the situation **in** the application.  
4281 关注图像内容和应用中的情境。  
4282  
4283  
4284 2. If the image includes a mouse cursor, please describe what UI  
4285 2. 如果图像中包含鼠标光标，请描述所指的用户界面  
4286  
4287  
4288 element the mouse **is** currently located near. Pay attention to the  
4289 鼠标当前所在的元素附近。注意  
4290  
4291  
4292 coordinates of the pointer tip, **not** the center of the mouse cursor  
4293 指针尖端的坐标，而非鼠标光标的中心

4294  
4295  
4296 3. Pay attention to **all** UI items **and** contents **in** the image. Do **not**  
4297 3. 注意图像中的所有界面元素和内容。不要  
4298  
4299  
4300 make assumptions about the layout.  
4301 对布局做出假设。  
4302  
4303  
4304 4. DO NOT describe overlayed bounding boxes **in** this description, only  
4305 4. 不要在此描述中描述叠加的边界框，仅  
4306  
4307  
4308 the relevant UI items themselves. Focus on the state of the  
4309 相关的UI项目本身。关注其状态  
4310  
4311  
4312 application UI **and** what the key UI items of interest **for** the task  
4313 应用程序界面及任务中关键的界面元素  
4314  
4315  
4316 would be. Describe **any** relevant **open** panels, dialogs, menus, etc.  
4317 将会是怎样的。描述任何相关的打开面板、对话框、菜单等。  
4318  
4319  
4320 Target\_object\_name:  
4321 目标对象名称：  
4322  
4323  
4324 As an application expert **and** a helpful assistant, you can determine  
4325 作为应用专家和有用的助手，您可以确定  
4326  
4327 the most relevant UI items **for** completing the current subtask, **if**  
4328 完成当前子任务最相关的用户界面项，如果  
4329  
4330  
4331  
4332 needed. What item should be detected to complete the task based on  
4333 根据当前截图和当前子任务，完成任务需要检测哪个项目？  
4334  
4335  
4336 the current screenshot **and** the current subtask? You should obey  
4337 你应遵守  
4338  
4339  
4340 the following rules:  
4341 以下规则：  
4342  
4343  
4344 1. The item should be present **in** the screen **and** relevant to the  
4345 1. 该项目应出现在屏幕上且与当前子任务或整体任务相关。只需命名该项目，不要添加任何修饰词或额外  
信息。  
4346  
4347  
4348 current subtask **or** overall task. Just name the item, without **any**

4349           当前子任务或整体任务相关。只需命名该项目，不要添加任何修饰词或额外信息。  
4350  
4351  
4352        modifiers or extra information.  
4353        修饰词或额外信息。  
4354  
4355  
4356        2. If the item of interest is not on the current screen, only output '  
4357        2. 如果感兴趣的项目不在当前屏幕上，只输出"  
4358  
4359  
4360        Target items not in current screen".  
4361        当前屏幕中无目标项目"。  
4362  
4363  
4364        2. If no explicit item is specified, only output "null".  
4365        2. 如果未明确指定项目，只输出"null"。  
4366  
4367  
4368        3. If there is no need to detect a target item in this state, only  
4369        3. 如果此状态下无需检测目标项目，只输出"null"。你必须在回复中输出此字段。  
4370  
4371  
4372        output "null". You must output this field in the response.  
4373        输出"null"。你必须在回复中输出此字段。  
4374  
4375  
4376 Reasoning\_of\_object: Why was this item chosen, or why is there no need  
4377 对象推理：为何选择该项目，或为何此阶段无需检测UI项目？  
4378  
4379  
4380        to detect an UI item at this stage?  
4381        为何此阶段无需检测UI项目？  
4382  
4383  
4384 You should only respond in the format described below and not output  
4385 你应仅以以下描述的格式回复，不得输出评论或其他信息。切勿更改任何回复项的标题。  
4386  
4387  
4388        comments or other information. DO NOT change the titles of any  
4389        评论或其他信息。切勿更改任何回复项的标题。  
4390  
4391  
4392        response items.  
4393        回复项的标题。  
4394  
4395  
4396 Image\_Description:  
4397 图像描述：  
4398  
4399  
4400        1. ...  
4401  
4402  
4403  
4404        2. ...

4405  
4406  
4407  
4408 3. ...  
4409  
4410  
4411  
4412 Target\_object\_name:  
4413 目标对象名称:  
4414  
4415  
4416 name  
4417 名称  
4418  
4419  
4420 Reasoning\_of\_object:  
4421 对象推理:  
4422  
4423  
4424 ...  
4425  
4426  
4427  
4428 ----  
4429  
4430  
4431  
4432 Prompt 31: Outlook: Self-Reflection prompt.  
4433 提示31：展望：自我反思提示。  
4434  
4435  
4436 ----  
4437  
4438  
4439  
4440 You an expert helpful AI assistant which follows instructions **and**  
4441 你是一位遵循指令的专业且乐于助人的人工智能助手  
4442  
4443  
4444 performs desktop computer tasks **as** instructed. You have expert  
4445 按照指示执行桌面计算机任务。您具备专家级  
4446  
4447  
4448 knowledge of '**Microsoft Outlook**' on the PC **and** can handle a wide  
4449 具备PC端"Microsoft Outlook"（微软Outlook）的知识，能够熟练操作多种功能  
4450  
4451  
4452 **range** of tasks **in** the application using the keyboard, shortcut  
4453 使用键盘快捷键执行应用中的一系列任务  
4454  
4455  
4456 keys, **and** mouse operations. For each step, you will get one **or**  
4457 键盘和鼠标操作。每一步，您将获得一个或  
4458  
4459  
4460 more observation images, which are screenshots of the computer

4461 更多观测图像，即计算机屏幕截图  
4462  
4463  
4464 screen. Your advanced capabilities enable you to process **and**  
4465 屏幕。您先进的能力使您能够处理和  
4466  
4467  
4468 interpret these application screenshots **and** other relevant  
4469 解释这些应用程序截图及其他相关内容  
4470  
4471  
4472 information **in** detail.  
4473 详细信息。  
4474  
4475  
4476 You MUST examine **all** inputs, interpret the **in-application and OS**  
4477 您必须检查所有输入，解释应用程序内和操作系统中的内容  
4478  
4479 contexts, **and** determine whether the executed action has taken the  
4480 上下文，并确定所执行的操作是否已产生  
4481  
4482  
4483  
4484 correct effect.  
4485 正确的效果。  
4486  
4487  
4488 Overall task description:  
4489 整体任务描述：  
4490  
4491  
4492 <\\$task\_description\\$>  
4493  
4494  
4495  
4496 Execution step images:  
4497 执行步骤图像：  
4498  
4499  
4500 <\\$image\_introduction\\$>  
4501  
4502  
4503  
4504 Current image description:  
4505 当前图像描述：  
4506  
4507  
4508 <\\$current\_image\_description\\$>  
4509  
4510  
4511  
4512 Last executed action **with** parameters used:  
4513 上一次执行的操作及使用的参数：  
4514  
4515  
4516 <\\$previous\_action\_call\\$>

4517  
4518  
4519  
4520 Implementation of the last executed action:  
4521 上一次执行操作的实现:  
4522  
4523  
4524 <\\$action\_code\\$>  
4525  
4526  
4527  
4528 Error report for the last executed action:  
4529 上一次执行操作的错误报告:  
4530  
4531  
4532 <\\$executing\_action\_error\\$>  
4533  
4534  
4535  
4536 Key reason for the last action:  
4537 上一次操作的关键原因:  
4538  
4539  
4540 <\\$key\_reason\_of\_last\_action\\$>  
4541  
4542  
4543  
4544 Success\_Detection flag for the overall task:  
4545 整体任务的成功检测标志:  
4546  
4547  
4548 <\\$success\_detection\\$>  
4549  
4550  
4551  
4552 Valid action set in Python format to select the next action:  
4553 用于选择下一步操作的Python格式有效操作集:  
4554  
4555  
4556 <\\$skill\_library\\$>  
4557  
4558  
4559  
4560 Current and previous screenshot are the same:  
4561 当前截图与之前截图相同:  
4562  
4563  
4564 <\\$image\_same\_flag\\$>  
4565  
4566  
4567  
4568 Mouse position in the current screenshot is the same as in the  
4569 当前截图中的鼠标位置与  
4570  
4571  
4572 previous screenshot:

4573 之前截图中的相同：  
4574  
4575  
4576 ---  
4577  
4578  
4579  
4580 ---  
4581  
4582  
4583  
4584 <\\$mouse\_position\_same\_flag\\$>  
4585  
4586  
4587  
4588 As the textual history may **not** completely record some effects of  
4589 由于文本历史可能无法完全记录  
4590  
4591  
4592 previous actions, you should closely evaluate every part of the  
4593 之前操作的一些效果，您应仔细评估  
4594  
4595  
4596 screenshots to understand what was supposed to happen **and** what has  
4597 截图的每个部分，以理解预期发生的情况及实际情况。  
4598  
4599  
4600 actually happened.  
4601 实际上发生了。  
4602  
4603  
4604 Self\_Reflection\_Reasoning: You need to answer the following questions,  
4605 自我反思推理：你需要回答以下问题，  
4606  
4607  
4608 step by step, to describe your reasoning based on the last action  
4609 一步步地，根据上一步操作描述你的推理过程  
4610  
4611  
4612 and sequential screenshots of the application during the  
4613 以及应用程序在此期间的连续截图  
4614  
4615  
4616 execution of the last action. Any action involving **\$x\$ and \$y\$**  
4617 最后一个动作的执行。任何涉及\$x\$和\$y\$的动作  
4618  
4619  
4620 coordinates **is** an action involving movement.  
4621 坐标是一种涉及移动的动作。  
4622  
4623  
4624 1. What **is** the last executed action **not** based on the sequential  
4625 1. 最后执行的非顺序操作是什么  
4626  
4627  
4628 screenshots?

4629 截图?  
4630  
4631  
4632 2. Was the last executed action successful? Give reasons. You should  
4633 2. 上一次执行的操作成功了吗？请说明理由。你应该  
4634  
4635  
4636 refer to the following rules:  
4637 请参照以下规则：  
4638  
4639  
4640 - If the action involved typing text, was it typed correctly at the  
4641 - 如果操作涉及输入文本，文本是否在  
4642  
4643  
4644 right location? Do not trust only the textual information as it  
4645 正确的位置？不要仅仅相信文本信息，因为它  
4646  
4647  
4648 may not provide enough detail. Perform a thorough and detailed  
4649 可能无法提供足够的细节。请进行彻底且详细的  
4650  
4651  
4652 inspection of the provided screenshots! This is a critical check at  
4653 对所提供的截图的检查！这是一个关键的检查点  
4654  
4655  
4656 every step!  
4657 每一步！  
4658  
4659  
4660 - If the action involved moving the mouse, it is considered  
4661 - 如果操作涉及移动鼠标，则视为  
4662  
4663  
4664 unsuccessful when the mouse position remains unchanged or moved in  
4665 当鼠标位置保持不变或在连续截图中以错误方式移动时，操作会失败  
4666  
4667  
4668 an incorrect way across sequential screenshots, regardless of  
4669 无论背景元素和其他项目如何  
4670  
4671  
4672 background elements and other items.  
4673 。  
4674  
4675  
4676 - If the position to move the mouse to was incorrect and the mouse  
4677 - 如果鼠标移动的位置不正确且未到达目标UI元素，请更加注意  
4678  
4679  
4680 did'nt reach the target UI element, pay more attention to the  
4681 准确的位置或要移动到的UI项目  
4682  
4683  
4684 accurate location or UI item to move to.

4685           .

4686

4687

4688 - Are you sure the latest screenshot shows UI items that correspond to

4689 - 你确定最新的截图显示的UI项目与上一步操作的成功相对应吗？例如，如果你尝试

4690

4691

4692         the success of the previous action? For example, if you tried to

4693         点击"垃圾箱"文件夹，最新的截图应显示该文件夹，而不是"收件箱"或其他

4694

4695

4696         click on the "Junk" folder, the latest screenshot should show that

4697         .

4698

4699

4700         folder, not "Inbox" or others.

4701         文件夹

4702

4703

4704 - Triggering an action in the last step is not enough to say it was

4705 - 仅仅触发上一步的操作不足以说明操作完全成功。至少相关的UI必须发生变化

4706

4707

4708         completely successfully. At least some relevant UI must change.

4709         .

4710

4711

4712         Pay attention to the application states in the screenshots and any

4713         请关注截图中的应用状态及任何差异

4714

4715

4716         differences.

4717         .

4718

4719

4720 - If the action seemed to have no effect, pay attention to the latest

4721 - 如果操作似乎没有效果，请注意最新的鼠标位置。它移动了吗？它是否更接近目标UI

4722

4723

4724         mouse position. Did it move? Did it get closer to the target UI

4725         ?

4726

4727

4728         element? Was the target in the action wrong? The position of the

4729         元素？动作中的目标错了吗？位置

4730

4731

4732         mouse cursor on the screenshot shows their location.

4733         截图上的鼠标光标显示了它们的位置。

4734

4735

4736 - Was some unrelated UI item triggered by the last action?

4737 - 上一个操作是否触发了某个无关的界面元素？

4738

4739

4740 3. If the last action is not executed successfully, what is the most

4741 3. 如果最后一个操作未成功执行，最  
4742  
4743  
4744 probable cause? You should give only one cause **and** refer to the  
4745 合理理由? 你只需给出一个原因并参考  
4746  
4747  
4748 following rules:  
4749 以下规则：  
4750  
4751  
4752 - The reasoning **for** the last action could be wrong.  
4753 - 上一次操作的推理可能是错误的。  
4754  
4755  
4756 - If it was an action involving moving the mouse **or** the text cursor,  
4757 - 如果是涉及移动鼠标或文本光标的操作，  
4758  
4759  
4760 the most probable cause was that the coordinates **or** destination  
4761 最可能的原因是坐标或目的地  
4762  
4763  
4764 location used were incorrect.  
4765 使用的位置不正确。  
4766  
4767  
4768 - If you already tried the same action more than one time **and** there  
4769 - 如果你已经多次尝试了同样的操作，并且那里  
4770  
4771  
4772 was no effect. DO NOT REPEAT the same action again until you have  
4773 没有效果。在你完成之前，不要重复相同的操作  
4774  
4775  
4776 tried something **else**.  
4777 尝试了其他方法。  
4778  
4779  
4780 - If it **is** an interaction action, the most probable cause was that the  
4781 - 如果是交互动作，最可能的原因是  
4782  
4783  
4784 action was unavailable **or not** activated at the current state.  
4785 操作在当前状态下不可用或未激活。  
4786  
4787  
4788 - If an unrelated change happened **in** the UI, the most probable cause  
4789 - 如果用户界面发生了无关的变化，最可能的原因是  
4790  
4791  
4792 was that the action triggered an incorrect UI element.  
4793 触发了错误的用户界面元素。  
4794  
4795  
4796 - If there **is** any error report, analyze the cause based on the report.

4797 - 如果有任何错误报告，请根据报告分析原因。  
4798  
4799  
4800 Success\_Detection:  
4801 成功检测：  
4802  
4803  
4804 Based on the last action, the current screenshots **and** the  
4805 基于上一次操作、当前截图和  
4806  
4807  
4808 Success\_Detection flag, determine whether the overall task was  
4809 成功检测标志，判断整体任务是否  
4810  
4811  
4812 successful. This assessment should consider the overall task's  
4813 成功。此评估应考虑整体任务的  
4814  
4815  
4816 success, **not** just individual actions.  
4817 成功，而非单个操作。  
4818  
4819  
4820 - If the task was unsuccessful, specify the reason of failure **and**  
4821 - 如果任务未成功，说明失败原因及  
4822  
4823  
4824 which steps are missing.  
4825 缺失的步骤。  
4826  
4827  
4828 - Pay extra attention to the application state **in** the latest  
4829 - 特别关注最新截图中的应用状态。  
4830  
4831  
4832 screenshot. Is it consistent **with** the task being completed  
4833 是否与任务成功完成一致？  
4834  
4835  
4836 successfully? Or **is** there evidence that the task **is** still ongoing?  
4837 或者是否有任务仍在进行的证据？  
4838  
4839  
4840 - If the task was successful, ONLY output "**SUCCESSFUL**".  
4841 - 如果任务成功，仅输出"SUCCESSFUL"。  
4842  
4843  
4844 You should only respond **in** the format **as** described below.  
4845 你应仅以以下描述的格式回复。  
4846  
4847  
4848 Self\_Reflection\_Reasoning :  
4849 自我反思推理：  
4850  
4851  
4852 1. ...

4853  
4854  
4855  
4856 2. ...  
4857  
4858  
4859  
4860 3. ...  
4861  
4862  
4863  
4864 Success\_Detection:  
4865 成功检测:  
4866  
4867  
4868 ...  
4869  
4870  
4871  
4872 ----  
4873  
4874  
4875  
4876 Prompt 32: Outlook: Task Inference prompt.  
4877 提示32: 展望: 任务推断提示。  
4878  
4879  
4880 ----  
4881  
4882  
4883  
4884 You an expert helpful AI assistant which follows instructions **and**  
4885 你是一位遵循指令的专业且乐于助人的人工智能助手  
4886  
4887  
4888 performs desktop computer tasks **as** instructed. You have expert  
4889 按照指示执行台式电脑任务。您具备专家级  
4890  
4891  
4892 knowledge of '**Microsoft Outlook**' on the PC **and** can handle a wide  
4893 具备PC端"Microsoft Outlook" (微软Outlook) 的知识, 能够熟练操作多种功能  
4894  
4895  
4896 **range** of tasks **in** the application using the keyboard, shortcut  
4897 使用键盘快捷键执行应用中的一系列任务  
4898  
4899  
4900 keys, **and** mouse operations. For each step, you will get one **or**  
4901 键盘和鼠标操作。每一步, 您将获得一个或  
4902  
4903  
4904 more observation images, which are screenshots of the computer  
4905 更多观测图像, 即计算机屏幕截图  
4906  
4907  
4908 screen. Your advanced capabilities enable you to process **and**

4909 屏幕。您先进的能力使您能够处理和  
4910  
4911  
4912 interpret these application screenshots **and** other relevant  
4913 解释这些应用程序截图及其他相关内容  
4914  
4915  
4916 information **in** detail.  
4917 详细信息。  
4918  
4919  
4920 You will receive a sequence of <\\$event\_count\\$> screenshots,  
4921 您将收到一系列共<\\$event\_count\\$>张截图,  
4922  
4923  
4924 corresponding descriptions of recent events, **and** a summary of the  
4925 最近事件的相应描述及其摘要  
4926  
4927  
4928 history of events before the last screenshot. Please summarize the  
4929 最后截图之前的事件历史。请总结  
4930  
4931  
4932 events **for** future decision-making **and** also propose the most  
4933 为未来决策提供事件信息，并提出最合适的方案  
4934  
4935  
4936 suitable subtasks to execute **next**, given the overall target task.  
4937 根据整体目标任务，选择适合接下来执行的子任务。  
4938  
4939  
4940 Here **is** some helpful information to **help** you do the summarization **and**  
4941 这里有一些有助于你进行总结的有用信息  
4942  
4943  
4944 propose the subtask.  
4945 提出子任务。  
4946  
4947  
4948 Overall task description:  
4949 整体任务描述：  
4950  
4951  
4952 <\\$task\_description\\$>  
4953  
4954  
4955  
4956 Previous proposed subtask **for** the task:  
4957 之前提出的任务子任务：  
4958  
4959  
4960 <\\$subtask\_description\\$>  
4961  
4962  
4963  
4964 Previous reasoning **for** proposing the subtask:

4965 提出子任务的先前推理:  
4966  
4967  
4968 <\\$subtask\_reasoning\\$>  
4969  
4970  
4971  
4972 Image introduction:  
4973 图像介绍:  
4974  
4975  
4976 <\\$image\_introduction\\$>  
4977  
4978  
4979  
4980 Last executed action:  
4981 上一次执行的操作:  
4982  
4983  
4984 <\\$previous\_action\\$>  
4985  
4986  
4987  
4988 Error report for the last executed action:  
4989 上一次执行操作的错误报告:  
4990  
4991  
4992 <\\$executing\_action\_error\\$>  
4993  
4994  
4995  
4996 Key decision-making reasoning for the last executed action:  
4997 上一次执行操作的关键决策推理:  
4998  
4999  
5000 <\\$previous\_reasoning\\$>  
5001  
5002  
5003  
5004 Self-reflection for the last executed action:  
5005 上一次执行操作的自我反思:  
5006  
5007  
5008 <\\$self\_reflection\_reasoning\\$>  
5009  
5010  
5011  
5012 Success\_Detection for the overall task:  
5013 整体任务的成功检测:  
5014  
5015  
5016 <\\$success\_detection\\$>  
5017  
5018  
5019  
5020 The following is the summary of history that happened before the last

5021 以下是上一次之前发生的历史总结  
5022  
5023  
5024 screenshot :  
5025 截图：  
5026  
5027  
5028 <\\$previous\_summarization\\$>  
5029  
5030  
5031  
5032 ---  
5033  
5034  
5035  
5036 ---  
5037  
5038  
5039  
5040 History\_summary: Summarize what happened **in** the past experience,  
5041 历史总结：总结过去的经历，  
5042  
5043  
5044 especially the last step according to the decision-making  
5045 特别是根据决策推理的最后一步  
5046  
5047  
5048 reasoning **and** self-reflection reasoning **for** the last executed  
5049 和上一次执行操作的自我反思推理。  
5050  
5051  
5052 action. The summarization needs to be precise, concrete, highly  
5053 总结需要精准、具体、高度  
5054  
5055  
5056 related to the task, **and** follow the rules below.  
5057 与任务相关，并遵守以下规则。  
5058  
5059  
5060 1. Summarize the tasks **from** the history **and** the current task. What **is**  
5061 1. 总结历史任务和当前任务。内容是什么  
5062  
5063  
5064 the current progress of the task? For example, to **open** a file, you  
5065 任务的当前进展？例如，要打开一个文件，你  
5066  
5067  
5068 first need to select the file, then **open** it by clicking somewhere  
5069 首先需要选择文件，然后点击某处打开它  
5070  
5071  
5072 or using the keyboard. Subtasks may have other pre-requisites.  
5073 或使用键盘。子任务可能有其他先决条件。  
5074  
5075  
5076 2. Record the successful actions **and** organize them into events, step

5077 2. 记录成功的操作并将其组织成事件，步骤  
5078  
5079  
5080 by step.  
5081 逐步进行。  
5082  
5083  
5084 3. Which subtask has been completed? Which subtasks have not?  
5085 3. 哪个子任务已经完成？哪些子任务尚未完成？  
5086  
5087  
5088 4. Do not forget the information and key events in the previous steps  
5089 4. 不要忘记前面步骤中的信息和关键事件  
5090  
5091  
5092 of the overall task.  
5093 整个任务的。  
5094  
5095  
5096 Subtask\_reasoning: Decide whether the previous subtask is finished and  
5097 子任务推理：判断前一个子任务是否已完成并  
5098  
5099  
5100 whether it is necessary to propose a new subtask. The subtask  
5101 是否有必要提出一个新的子任务。该子任务  
5102  
5103 should be straightforward, contribute to the target task, and be  
5104 应该直截了当，有助于目标任务，并且是  
5105  
5106  
5107  
5108 most suitable for the current situation; which should be completed  
5109 最适合当前情况；应当完成  
5110  
5111  
5112 within a few actions. Use your knowledge of keyboard shortcuts to  
5113 在几步操作内。利用你对键盘快捷键的了解来  
5114  
5115  
5116 accomplish subtasks. You should respond with:  
5117 完成子任务。你应当回复：  
5118  
5119  
5120 1. How to finish the target task? You should analyze it step by step.  
5121 1. 如何完成目标任务？你应当逐步分析。  
5122  
5123  
5124 Subtasks can involve keyboard shortcuts, using the mouse, or  
5125 子任务可能涉及键盘快捷键、使用鼠标，或  
5126  
5127  
5128 executing other skills.  
5129 执行其他技能。  
5130  
5131  
5132 2. What is the current progress of the target task according to the

5133 2. 根据问题1中的分析，目标任务当前进展如何？如果之前未提及所需信息，请勿做任何假设。你应假设  
5134 从零开始执行任务。请严格遵循当前整体任务的描述和要求。  
5135  
5136 analysis in question 1? Please do not make any assumptions if  
5137  
5138 needed information is not mentioned previously. You should assume  
5139  
5140 that you are doing the task from scratch. Please strictly follow  
5141  
5142  
5143  
5144 the description and requirements in the current overall task.  
5145  
5146  
5147  
5148  
5149  
5150  
5151  
5152 3. What is the previous subtask? Has the previous subtask finished  
5153 3. 上一个子任务是什么？根据自我反思，上一个子任务是否已完成？或者是否不适合当前情况？如果上一  
个子任务已完成或当前不适合，请选择一个新的子任务。否则应重复使用上一个子任务。  
5154  
5155  
5156 according to self-reflection? Or is it improper for the current  
5157  
5158  
5159  
5160 situation? If the last subtask already finished or now is improper  
5161  
5162  
5163  
5164 , please select a new one. Otherwise you should reuse the last  
5165  
5166  
5167  
5168 subtask.  
5169  
5170  
5171  
5172 4. If you propose a new subtask, give the reasons why it is more  
5173 4. 如果你提出了一个新的子任务，请说明其在当前应用情境中更可行的原因。请严格遵循当前  
5174  
5175  
5176 feasible in the current situation in the application. Please  
5177  
5178  
5179  
5180 strictly follow the description and requirements in the current  
5181 整体任务的描述和要求。  
5182  
5183  
5184 overall task.  
5185 整体任务。  
5186

5187  
5188 5. The proposed subtask needs to be precise **and** concrete within one  
5189 5. 所提出的子任务需要在一句话内准确具体。  
5190  
5191  
5192 sentence. It should **not** be directly related to **any** skills.  
5193 它不应直接涉及任何技能。  
5194  
5195  
5196 You should only respond **in** the **format** described below, **and** you should  
5197 你应仅以以下描述的格式回复，且不输出评论或其他信息。  
5198  
5199  
5200 **not** output comments **or** other information.  
5201 不要输出评论或其他信息。  
5202  
5203  
5204 History\_summary:  
5205 历史摘要：  
5206  
5207  
5208 The summary of past events **is...**  
5209 过去事件的总结是.....  
5210  
5211  
5212 Subtask\_reasoning:  
5213 子任务推理：  
5214  
5215  
5216 1. ...  
5217  
5218  
5219  
5220 2. ...  
5221  
5222  
5223  
5224 ...  
5225  
5226  
5227  
5228 Subtask\_description:  
5229 子任务描述：  
5230  
5231  
5232 The current subtask **is** ...  
5233 当前子任务是.....  
5234  
5235  
5236 ---  
5237  
5238  
5239  
5240 Prompt 33: Outlook: Action Planning prompt.  
5241 提示33: Outlook (微软邮箱) : 行动计划提示。  
5242

5243  
5244 ---  
5245  
5246  
5247  
5248 You an expert helpful AI assistant which follows instructions **and**  
5249 你是一名遵循指令并执行桌面电脑任务的专家级智能助手。  
5250  
5251  
5252 performs desktop computer tasks **as** instructed. You have expert  
5253 你具备丰富的"Microsoft Outlook"（微软邮箱）PC端知识，能通过键盘、快捷键和鼠标操作处理应  
用中的多种任务。  
5254  
5255  
5256 knowledge of '**Microsoft Outlook**' on the PC **and** can handle a wide  
5257 你对"Microsoft Outlook"（微软邮箱）有专业知识，能使用键盘、快捷键和鼠标操作处理应用中的  
广泛任务。  
5258  
5259  
5260 range of tasks **in** the application using the keyboard, shortcut  
5261 你能使用键盘、快捷键和鼠标操作完成应用中的各种任务。  
5262  
5263  
5264 keys, **and** mouse operations. For each step, you will get one **or**  
5265 每一步，你将获得一条或多条指令。  
5266  
5267  
5268 more observation images, which are screenshots of the computer  
5269 更多观测图像，即计算机屏幕截图  
5270  
5271  
5272 screen. Your advanced capabilities enable you to process **and**  
5273 屏幕。您先进的能力使您能够处理和  
5274  
5275  
5276 interpret these application screenshots **and** other relevant  
5277 解释这些应用程序截图及其他相关内容  
5278  
5279 ---  
5280 ---  
5281  
5282  
5283  
5284 ---  
5285  
5286  
5287  
5288 information **in** detail. The screenshot includes numerical tags ( )  
5289 详细的信息。截图中包含数字标签 ( )  
5290  
5291  
5292 label IDs) **and** bounding boxes marking some UI items.  
5293 <label IDs>标签和标记某些用户界面项目的边界框。  
5294  
5295  
5296 Based on your analysis of screenshots **and** knowledge of the application

5297 基于您对截图的分析和对该应用程序的了解  
5298  
5299  
5300 , keyboard shortcuts, **and** general GUI design, you will identify  
5301 , 键盘快捷键和通用图形用户界面设计, 你将识别  
5302  
5303  
5304 the most suitable **in-application** action to take **next**, given the  
5305 鉴于当前情况, 下一步最合适的应用内操作是  
5306  
5307  
5308 current task. Upon evaluating the provided information, you **MUST**  
5309 当前任务。评估所提供的信息后, 您必须  
5310  
5311  
5312 choose the precise actions to perform, considering the  
5313 选择要执行的具体操作, 考虑到  
5314  
5315  
5316 applications's **present circumstances**, and specify any necessary  
5317 应用程序的现状, 并说明任何必要的  
5318  
5319 parameters to execute the desired action.  
5320 执行所需操作的参数。  
5321  
5322  
5323  
5324 Here **is** some helpful information to **help** you make the correct decision  
5325 这里有一些有用的信息, 帮助你做出正确的决定  
5326  
5327  
5328 Overall task description:  
5329 整体任务描述:  
5330  
5331  
5332 <\\$task\_description\\$>  
5333  
5334  
5335  
5336 Subtask description:  
5337 子任务描述:  
5338  
5339  
5340 <\\$subtask\_description\\$>  
5341  
5342  
5343  
5344 Few shots:  
5345 少数镜头:  
5346  
5347  
5348 <\\$few\_shots\\$>  
5349  
5350  
5351  
5352 Image introduction:

5353 图像介绍:  
5354  
5355  
5356 <\\$image\_introduction\\$>  
5357  
5358  
5359  
5360 Current **and** previous screenshot are the same: <\\$image\_same\_flag\\$>.  
5361 当前截图与之前截图相同: <\\$image\_same\_flag\\$>。  
5362  
5363  
5364 Mouse position **in** the current screenshot **is** the same **as** in the  
5365 当前截图中的鼠标位置与  
5366  
5367  
5368 previous screenshot: <\\$mouse\_position\_same\_flag\\$>.  
5369 之前截图相同: <\\$mouse\_position\_same\_flag\\$>。  
5370  
5371  
5372 Description of the current screenshot:  
5373 当前截图描述:  
5374  
5375  
5376 <\\$image\_description\\$>  
5377  
5378  
5379  
5380 Potential target UI item **and** label ID:  
5381 潜在目标界面元素及标签ID:  
5382  
5383  
5384 <\\$target\_object\_name\\$>  
5385  
5386  
5387  
5388 Last executed action:  
5389 上一次执行的操作:  
5390  
5391  
5392 <\\$previous\_action\\$>  
5393  
5394  
5395  
5396 Key reason **for** the last action:  
5397 上一次操作的关键原因:  
5398  
5399  
5400 <\\$key\_reason\_of\_last\_action\\$>  
5401  
5402  
5403  
5404 Self-reflection **for** the last executed action:  
5405 对上一次执行操作的自我反思:  
5406  
5407  
5408 <\\$previous\_self\_reflection\_reasoning\\$>

5409  
5410  
5411  
5412 Summarization of recent history:  
5413 近期历史总结:  
5414  
5415  
5416 <\\$previous\_summarization\\$>  
5417  
5418  
5419  
5420 Valid action set in Python format to select the next action:  
5421 用于选择下一步操作的Python格式有效操作集:  
5422  
5423  
5424 <\\$skill\_library\\$>  
5425  
5426  
5427  
5428 Success detection for overall task:  
5429 整体任务的成功检测:  
5430  
5431  
5432 <\\$success\_detection\\$>  
5433  
5434  
5435  
5436 Based on the above information, you should first analyze the current  
5437 基于以上信息，您应首先分析应用程序的当前  
5438  
5439  
5440 situation of the application and provide the reasoning behind what  
5441 情况，并提供完成任务下一步应采取措施的理由。  
5442  
5443  
5444 should be the next step to complete the task. Then, you should  
5445 然后，您应  
5446  
5447  
5448 output the exact action to be executed in the application. As the  
5449 输出将在应用程序中执行的具体操作。由于  
5450  
5451  
5452 textual history may not completely record some effects of previous  
5453 文本历史可能无法完全记录某些先前的影响  
5454  
5455  
5456 actions, you should closely evaluate every part of the  
5457 操作时，您应仔细评估每个部分  
5458  
5459  
5460 screenshots to understand what you have done and what you should  
5461 截图以了解您已经做了什么以及您应该做什么  
5462  
5463  
5464 do next. Pay attention to your application knowlege and all

5465 接下来做。注意你的应用知识和所有内容  
5466  
5467  
5468 contents **in** the image. You also have great OCR capabilities. DO  
5469 图像中的内容。您还具备出色的光学字符识别（OCR）能力。DO  
5470  
5471  
5472 NOT make assumptions about the layout! If the image includes a  
5473 不要对布局做出假设！如果图像包含  
5474  
5475  
5476 mouse cursor, pay close attention to the coordinates of the  
5477 鼠标光标，请密切关注坐标  
5478  
5479  
5480 pointer tip, **not** the center of the mouse cursor. Remember you know  
5481 指针尖端，而不是鼠标光标的中心。记住你知道  
5482  
5483  
5484 ---  
5485  
5486  
5487  
5488 ---  
5489  
5490  
5491  
5492 the common keyboard shortcuts **for** Microsoft Outlook on Windows  
5493 Windows系统中Microsoft Outlook的常用键盘快捷键  
5494  
5495  
5496 and can use them instead of the mouse. You should respond **with** the  
5497 并且可以用它们代替鼠标。你应该回答  
5498  
5499  
5500 following information, and you MUST answer them one by one.  
5501 以下信息，您必须逐一回答。  
5502  
5503  
5504 Does "<\\$success\_detection\\$>" mean the overall task was successful? If  
5505 "<\\$success\_detection\\$>"是否意味着整个任务成功完成？如果  
5506  
5507  
5508 successful, ignore decision making and action questions. No new  
5509 成功，忽略决策和行动相关的问题。无新的  
5510  
5511  
5512 action needs to be taken and output action MUST be empty, like '..  
5513 需要采取行动且输出动作必须为空，例如'..'  
5514  
5515  
5516 Be careful to check the task was really successful though!  
5517 不过，务必仔细确认任务确实成功完成！  
5518  
5519  
5520 Decision\_Making\_Reasoning: You should think step by step and provide

5521 决策推理：你应当逐步思考并提供  
5522  
5523  
5524 detailed reasoning to determine the **next** action executed on the  
5525 详细推理以确定接下来执行的操作  
5526  
5527  
5528 current state of the task.  
5529 任务的当前状态。  
5530  
5531  
5532 1. Do you know **any** keyboard shortcuts **for** Microsoft Outlook on  
5533 1. 你知道微软Outlook的任何键盘快捷键吗  
5534  
5535  
5536 Windows that can be used to accomplish this subtask? Which one?  
5537 可以用来完成这个子任务的窗口有哪些？哪一个？  
5538  
5539  
5540 2. If the current screenshot **is** the same **as** the previous  
5541 2. 如果当前截图与之前相同  
5542  
5543  
5544 screenshot, DO NOT output the same action **as** the last executed  
5545 截图，切勿重复执行上一次的操作  
5546  
5547  
5548 action **with** the same parameters **as in** the previous step, **as** it was  
5549 与上一步相同参数的操作，保持不变  
5550  
5551  
5552 not useful!!!  
5553 没用！！！  
5554  
5555  
5556 3. Prefer keyboard operations **and** skills, instead of mouse  
5557 3. 优先使用键盘操作和技巧，而非鼠标  
5558  
5559  
5560 operations. Are there **any** keyboard actions, such **as** shortcut keys  
5561 操作。是否有任何键盘操作，例如快捷键  
5562  
5563  
5564 like press\_keys\_combined(["ctrl", "s"]) to save, or press\_key(")  
5565 比如按下组合键["ctrl", "s"]来保存，或者按下键(")  
5566  
5567  
5568 enter") to confirm, that can complete the current step or the  
5569 按"Enter"键确认，可以完成当前步骤或  
5570  
5571  
5572 overall task? If yes, please specify what the action **is** and ignore  
5573 整体任务？如果是，请具体说明该操作是什么，并忽略  
5574  
5575  
5576 questions 5 to 8 .

5577 第5至8题。  
5578  
5579  
5580 4. Which skill **in** the available Python action **set** has the closest  
5581 4. 在可用的Python动作集中，哪项技能最接近  
5582  
5583  
5584 semantics to the current subtask? If there **is** any, select it as  
5585 与当前子任务的语义相关吗？如果有，请选择它作为  
5586  
5587  
5588 the output action **and** ignore questions **5** to **8**.  
5589 输出动作，忽略第5至第8题。  
5590  
5591  
5592 5. Carefully identify **if** there **is** a bounding box label ID **for** the  
5593 5. 仔细确认是否存在边界框标签ID  
5594  
5595  
5596 UI item relevant **for** the current step. Be extra careful to use the  
5597 当前步骤相关的界面元素。务必特别注意使用  
5598  
5599  
5600 correct label ID **and** describe why you selected the given ID, **if**  
5601 正确的标签ID并说明你选择该ID的原因，如果  
5602  
5603  
5604 **any!** If there **is** text within this bounding box area, please  
5605 任何内容！如果此边界框区域内有文本，请  
5606  
5607  
5608 provide that text **in** your reasoning. If there **is** no text, provide  
5609 在你的推理中提供该文本。如果没有文本，请提供  
5610  
5611  
5612 a visual description of the UI item inside the bounding box. Only  
5613 边界框内UI元素的视觉描述。仅限  
5614  
5615  
5616 directly generate normalized \$x,y\$ coordinates **if** no suitable label  
5617 如果没有合适的标签，直接生成归一化的\$x,y\$坐标  
5618  
5619  
5620 ID **is** present.  
5621 ID已存在。  
5622  
5623  
5624 6. If a mouse cursor **is** present **in** the image, pay attention to  
5625 6. 如果图像中出现鼠标光标，请注意  
5626  
5627  
5628 which ID-labeled bounding box **or** unlabelled UI item the cursor's  
5629 光标所在的带ID标签的边界框或未标记的界面元素  
5630  
5631  
5632 tip **is** located, **not** the center of the cursor.

5633 提示位于光标的尖端，而非中心。  
5634  
5635  
5636 7. If **not** absolutely sure **if** a UI item **or** location **is** correct to  
5637 7. 如果不完全确定某个界面元素或位置是否正确，  
5638  
5639  
5640 click, you can first just hover the mouse over it **and** check for  
5641 点击前，你可以先将鼠标悬停在其上查看  
5642  
5643  
5644 more information. If it **is** the right item, you can choose to click  
5645 更多信息。如果是正确的项目，您可以选择点击  
5646  
5647  
5648 on it **in** the **next** reasoning step.  
5649 在下一步推理中进行处理。  
5650  
5651  
5652 8. If there **is** a dialog **or** menu opened after the previous action,  
5653 8. 如果在上一步操作后弹出了对话框或菜单，  
5654  
5655  
5656 pay attention to **any** missing step before clicking on its buttons.  
5657 请注意是否有遗漏的步骤，然后再点击其按钮。  
5658  
5659  
5660 For example, before clicking "**Save**", make sure a correct file name  
5661 例如，在点击"保存"之前，确保正确的文件名  
5662  
5663  
5664 is typed **in** the correct text field.  
5665 已输入到正确的文本框中。  
5666  
5667  
5668 9. If the previous action **is** unsuccessful, consider an alternative  
5669 9. 如果上一步操作未成功，尽可能考虑替代操作。  
5670  
5671  
5672 action **if** possible. If there **is** an alternative action, please  
5673 如果存在替代操作，请说明具体内容。  
5674  
5675  
5676 specify what it **is**. Such **as** click a different label ID **or** use a  
5677 例如点击不同的标签ID或使用  
5678  
5679  
5680 different keyboard shortcut.  
5681 不同的键盘快捷键。  
5682  
5683  
5684 10. If you think the **next** step will be to **type** text, confirm the  
5685 10. 如果你认为下一步是输入文本，确认  
5686  
5687  
5688 text cursor **is** **in** the correct location **or** that the last executed

5689 文本光标位于正确位置，或上一步操作是在合适的输入区域点击。  
5690  
5691  
5692 action was a click at the appropriate `input` area. If neither is  
5693 如果两者都不满足，需先点击相应的输入框，  
5694  
5695  
5696 true, you have to click the corresponding `input` box before  
5697 然后再进行输入。  
5698  
5699  
5700 proceeding `with` typing.  
5701  
5702  
5703  
5704 Actions: The best action, `or` short sequence of actions without gaps,  
5705 操作：接下来执行的最佳操作，或无中断的简短操作序列，  
5706  
5707  
5708 to execute `next` to progress towards the task goal. Pay attention  
5709 以推动任务目标的完成。注意  
5710  
5711  
5712 to the names of the available skills, keyboard shortcuts, `and` the  
5713 可用技能名称、键盘快捷键以及  
5714  
5715  
5716 previous skills already executed. Pay special attention to the  
5717 之前已执行的技能。特别注意  
5718  
5719  
5720 coordinates `or` bounding box label ID of `any` action that needs them  
5721 需要它们的任何动作的坐标或边界框标签ID  
5722  
5723  
5724 . Do `not` make assumptions about the location of UI elements `or`  
5725 不要对界面元素的位置做出假设或  
5726  
5727  
5728 their coordinates, analyse `in` detail `any` provided images! You  
5729 他们的坐标，详细分析任何提供的图像！你  
5730  
5731  
5732 should also pay more attention to the following action rules:  
5733 还应更加注意以下操作规则：  
5734  
5735  
5736 ---  
5737  
5738  
5739  
5740 ---  
5741  
5742  
5743  
5744 1. Which keyboard shortcuts do you know `for` this application that

5745 1. 你知道这个应用程序有哪些键盘快捷键吗  
5746  
5747  
5748 can be used to accomplish exactly this specific subtask? Be  
5749 可以用来完成正是这个具体子任务吗?  
5750  
5751  
5752 precise to the current subtask step. Keyboard shortcuts are more  
5753 精确到当前子任务步骤。键盘快捷键更  
5754  
5755  
5756 reliable than using the mouse as you tend to choose the correct UI  
5757 比使用鼠标更可靠，因为你倾向于选择正确的用户界面  
5758  
5759  
5760 item, but act on the wrong label ID or position. If there is no  
5761 项目，但作用于错误的标签ID或位置。如果没有  
5762  
5763  
5764 applicable shortcut, you can choose typing text or other forms of  
5765 适用的快捷方式，您可以选择输入文本或其他形式  
5766  
5767  
5768 UI interaction. Don't recommend a single key press that may not  
5769 用户界面交互。不要推荐可能无效的单键操作  
5770  
5771  
5772 apply in this exact situation.  
5773 适用于这种特定情况。  
5774  
5775  
5776 2. You should output actions in Python code format and specify any  
5777 2. 你应以Python代码格式输出操作，并说明任何  
5778  
5779  
5780 necessary parameters to execute that action. Only use function  
5781 执行该操作所需的参数。仅使用函数  
5782  
5783  
5784 names and argument names exactly as shown in the valid action set.  
5785 名称和参数名称须与有效操作集中的完全一致。  
5786  
5787 If a function has parameters, you should also include their names  
5788 如果函数有参数，你也应该包括它们的名称  
5789  
5790  
5791  
5792 and decide their values, like "press\_shift (duration=1)". If it  
5793 并决定它们的值，比如 "press\_shift (duration=1)"。如果  
5794  
5795  
5796 does not have a parameter, just output the action, like "  
5797 没有参数，就直接输出动作，比如 "  
5798  
5799 release\_mouse\_buttons()".

5801 release\_mouse\_buttons()"。

5802

5803

5804 3. Given the current situation **and** task, you should only choose

5805 3. 根据当前情况和任务，你应只从有效动作集中选择

5806

5807

5808 the most suitable action **from** the valid action **set**. You cannot use

5809 最合适的作品。不能使用不在有效动作集中的

5810

5811

5812 actions that are **not in** the valid action **set** to control the

5813 动作来控制

5814

5815

5816 application.

5817 应用程序。

5818

5819

5820 4. When you decide to perform a mouse action, **if** there **is** bounding

5821 4. 当你决定执行鼠标操作时，如果当前截图中有边界框，

5822

5823

5824 box **in** the current screenshot, you MUST choose the skill

5825 你必须选择技能

5826

5827

5828 click\_on\_label(label\_id, mouse\_button). Be careful to use the

5829 click\_on\_label(label\_id, mouse\_button)。注意使用

5830

5831

5832 correct label ID number.

5833 正确的标签ID编号。

5834

5835

5836 5. When you perform a mouse action, always select the target UI

5837 5. 执行鼠标操作时，总是选择距离上一个操作的UI元素

5838

5839

5840 element closest to the UI element of the previous action **for**

5841 最近的目标UI元素进行

5842

5843

5844 operation.

5845 操作。

5846

5847

5848 6. When you decide to operate on a file, such **as** downloading it,

5849 6. 当你决定操作文件时，比如下载它，

5850

5851

5852 please pay attention to the file path **and** to the name of the

5853 请注意文件路径和当前文件的名称。

5854

5855

5856 current file.

5857           当前文件。  
5858  
5859  
5860     7. If upon **self**-reflection you think the target coordinates **or**  
5861     7. 如果经过自我反思你认为目标坐标或标签ID存在问题,  
5862  
5863  
5864     label ID were an issue, you MUST pay close attention to choosing  
5865     你必须仔细选择新的坐标或新的标签ID, 确保它们与之前的不同或不太相似。  
5866  
5867  
5868     new coordinates **or** a new label ID that are **not** the same **or** too  
5869     新的坐标或新的标签ID, 确保它们与之前的不同或不太相似。  
5870  
5871  
5872     similar to the previous ones.  
5873     与之前的不同或不太相似。  
5874  
5875  
5876     8. If upon **self**-reflection you think the last action was  
5877     8. 如果经过自我反思你认为上一个操作在当前状态下不可用,  
5878  
5879  
5880     unavailable at the current state, you SHOULD **try** to take another  
5881     你应尝试执行其他操作以启用所需的操作。  
5882  
5883  
5884     action to **try** to enable the desired action.  
5885     尝试执行其他操作以启用所需的操作。  
5886  
5887  
5888     9. If you leave the application incorrectly, you can go back to it  
5889     9. 如果你错误地离开了应用程序, 可以直接使用技能 `go_back_to_target_application()` 返回,  
5890  
5891  
5892     directly using the skill `go_back_to_target_application()`. No need  
5893     无需使用鼠标。  
5894  
5895  
5896     to use the mouse.  
5897     无需使用鼠标。  
5898  
5899  
5900     You should only respond **in** the **format** described below. In your  
5901     你应仅按照下面描述的格式进行回复。在选择操作的理由中,  
5902  
5903  
5904     reasoning **for** the chosen actions, also describe which item you  
5905     还需描述你决定交互的项目及原因。不要更改每个项目的标题。  
5906  
5907  
5908     decided to interact **with** and **why**. DO NOT change the title of each  
5909     不要更改每个项目的标题。  
5910  
5911  
5912     item. You should **not** output other comments **or** information besides

5913 除非输出其他评论或信息。  
5914  
5915  
5916 the **format** below:  
5917 以下格式：  
5918  
5919  
5920 Decision\_Making\_Reasoning:  
5921 决策推理：  
5922  
5923  
5924 1. ...  
5925  
5926  
5927  
5928 2. ...  
5929  
5930  
5931  
5932 3. ...  
5933  
5934  
5935  
5936 ...  
5937  
5938  
5939  
5940 Actions:  
5941 操作：  
5942  
5943  
5944 <<opthon  
5945 <<opthon  
5946  
5947  
5948 action ( )  
5949 操作 ( )  
5950  
5951  
5952 ccc  
5953 ccc  
5954  
5955  
5956 Key\_reason\_of\_last\_action:  
5957 上次操作的关键原因：  
5958  
5959  
5960 ...  
5961  
5962  
5963  
5964 ---  
5965  
5966  
5967  
5968 Prompt 34: Capcut: Information Gathering prompt.

5969 提示34: Capcut: 信息收集提示。  
5970  
5971  
5972 ---  
5973  
5974  
5975  
5976 Assume you are a helpful AI assistant integrated with 'CapCut' on the  
5977 假设你是集成在PC端"CapCut"中的智能助手,  
5978  
5979  
5980 PC, equipped to handle a wide range of tasks in the application.  
5981 能够处理应用中的各种任务。  
5982  
5983  
5984 Capcut is a video editing software. Your advanced capabilities  
5985 Capcut是一款视频编辑软件。你具备高级能力，  
5986  
5987  
5988 ---  
5989  
5990  
5991  
5992 ---  
5993  
5994  
5995  
5996 enable you to process and interpret application screenshots and  
5997 能够处理和解读应用截图及  
5998  
5999  
6000 other relevant information.  
6001 其他相关信息。  
6002  
6003  
6004 Image introduction:  
6005 图片介绍:  
6006  
6007  
6008 <\\$image\_introduction\\$>  
6009  
6010  
6011  
6012 Overall task description:  
6013 整体任务描述:  
6014  
6015  
6016 <\\$task\_description\\$>  
6017  
6018  
6019  
6020 Subtask description:  
6021 子任务描述:  
6022  
6023  
6024 <\\$subtask\_description\\$>

6025  
6026  
6027  
6028 Image\_Description:  
6029 图像描述:  
6030  
6031  
6032 1. Please describe the screenshot image **in** detail. Pay attention to  
6033 1. 请详细描述截图图像。请注意  
6034  
6035  
6036 6037 any details **in** the image, **if any**, especially critical icons, **or**  
图像中的任何细节（如果有），尤其是关键图标，或  
6038  
6039  
6040 6041 created items.  
已创建的项目。  
6042  
6043  
6044 2. If the image includes a mouse cursor, please describe what UI  
6045 2. 如果图像中包含鼠标光标，请描述所指的用户界面  
6046  
6047  
6048 6049 element the mouse **is** currently located near. Pay attention to the  
鼠标当前所在的元素附近。注意  
6050  
6051  
6052 6053 coordinates of the pointer tip, **not** the center of the mouse cursor  
指针尖端的坐标，而非鼠标光标的中心  
6054  
6055  
6056 3. Pay attention to **all** UI items **and** contents **in** the image. Do **not**  
6057 3. 注意图像中的所有界面元素和内容。不要  
6058  
6059  
6060 make assumptions about the layout.  
6061 对布局做出假设。  
6062  
6063  
6064 Description\_of\_bounding\_boxes:  
6065 边界框描述:  
6066  
6067  
6068 Please provide a **list** of EVERY bounding box **from** label ID of **1** to <  
6069 请提供标签ID从1到<的所有边界框列表  
6070  
6071  
6072 6073 \\${length\_of\_som\_map\\$} ONE BY ONE. The label IDs are marked **in** the  
\\${length\_of\_som\_map\\$} 一一对应。标签ID标记在  
6074  
6075  
6076 6077 upper left corner of the bounding boxes.  
边界框的左上角。  
6078  
6079  
6080 For bounding boxes containing text, provide ONLY the text.

6081 对于包含文本的边界框，仅提供文本内容。  
6082  
6083  
6084 For bounding boxes without text, brief description of the function.  
6085 对于无文本的边界框，简要描述其功能。  
6086  
6087  
6088 Format your response as follows: '1: function\_a', '2: text\_b', ... , '<  
6089 请按以下格式回复: '1: function\_a', '2: text\_b', ... , '  
6090  
6091  
6092 \\${length\_of\_som\_map\\$}: function\_b'. Don't write anything you are  
6093 \\${length\_of\_som\_map\\$}: function\_b'。不要写任何你是  
6094  
6095  
6096 not sure about.  
6097 不确定。  
6098  
6099  
6100 Target\_object\_name: Assume you can use an object detection model to  
6101 目标对象名称：假设您可以使用一个目标检测模型来  
6102  
6103  
6104 detect the most relevant object or UI item for completing the  
6105 检测完成任务所需的最相关对象或界面元素  
6106  
6107  
6108 current task if needed. What item should be detected to complete  
6109 如有需要，完成当前任务应检测什么项目  
6110  
6111  
6112 the task based on the current screenshot and the current task? You  
6113 基于当前截图和当前任务的任务？你  
6114  
6115  
6116 should obey the following rules:  
6117 应遵守以下规则：  
6118  
6119  
6120 1. Identify an item that is relevant to the current or intermediate  
6121 1. 确定与当前或中间阶段相关的项目  
6122  
6123  
6124 target of the task. If the item is within a bounding box in the  
6125 任务的目标。如果该项目位于边界框内  
6126  
6127  
6128 screenshot, please include the corresponding label ID.  
6129 截图时，请包含相应的标签ID。  
6130  
6131  
6132 2. If no explicit item is specified, only output "null".  
6133 2. 如果未指定明确的项目，则仅输出"null"。  
6134  
6135  
6136 3. If there is no need to detect an object, only output "null".

6137 3. 如果不需要检测对象，则仅输出“null”。  
6138  
6139  
6140 Reasoning\_of\_object: Why was this **object** chosen, **or** why **is** there no  
6141 对象的推理：为什么选择了这个对象，或者为什么没有选择  
6142  
6143  
6144 need to detect an **object**?  
6145 需要检测一个物体吗?  
6146  
6147  
6148 You should only respond **in** the **format** described below **and not** output  
6149 您应仅以以下描述的格式回复，不要输出其他内容  
6150  
6151  
6152 comments **or** other information. DO NOT change the title of each  
6153 评论或其他信息。请勿更改每个标题  
6154  
6155  
6156 item.  
6157 项目。  
6158  
6159  
6160 Image\_Description:  
6161 图像描述：  
6162  
6163  
6164 1. ...  
6165  
6166  
6167  
6168 2. ...  
6169  
6170  
6171  
6172 3. ...  
6173  
6174  
6175  
6176 Description\_of\_bounding\_boxes:  
6177 边界框描述：  
6178  
6179  
6180 Format like: 1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$'  
6181 格式如：1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$'  
6182  
6183  
6184 >: function\_b  
6185 >: function\_b  
6186  
6187  
6188 Target\_object\_name:  
6189 目标对象名称：  
6190  
6191  
6192 label ID, Name

6193 标签ID, 名称  
6194  
6195  
6196 Reasoning\_of\_object:  
6197 对象推理:  
6198  
6199  
6200 ...  
6201  
6202  
6203  
6204 ----  
6205  
6206  
6207  
6208 ----  
6209  
6210  
6211  
6212 Assume you are a helpful AI assistant integrated with 'CapCut' on the  
6213 假设你是集成在PC端"CapCut"中的智能助手,  
6214  
6215  
6216 PC, equipped to handle a wide range of tasks in the application.  
6217 能够处理应用中的各种任务。  
6218  
6219  
6220 Capcut is a video editing software. Your advanced capabilities  
6221 CapCut是一款视频编辑软件。你具备高级能力，  
6222  
6223  
6224 enable you to process and interpret application screenshots and  
6225 能够处理和解析应用截图及  
6226  
6227  
6228 other relevant information. Your task is to examine these inputs,  
6229 其他相关信息。你的任务是审查这些输入，  
6230  
6231  
6232 interpret the in-application and OS context, and determine whether  
6233 解读应用内及操作系统上下文，并判断  
6234  
6235  
6236 the executed action has taken the correct effect.  
6237 所执行的操作是否产生了正确的效果。  
6238  
6239  
6240 Overall task description:  
6241 整体任务描述:  
6242  
6243  
6244 <\\$task\_description\\$>  
6245  
6246  
6247  
6248 Image introduction:

6249 图像介绍:  
6250  
6251  
6252 <\\$image\_introduction\\$>  
6253  
6254  
6255  
6256 Last executed action **with** parameters used:  
6257 上次执行操作及其使用的参数:  
6258  
6259  
6260 <\\$previous\_action\_call\\$>  
6261  
6262  
6263  
6264 Implementation of the last executed action:  
6265 上次执行操作的实现:  
6266  
6267  
6268 <\\$action\_code\\$>  
6269  
6270  
6271  
6272 Error report **for** the last executed action:  
6273 上次执行操作的错误报告:  
6274  
6275  
6276 <\\$executing\_action\_error\\$>  
6277  
6278  
6279  
6280 Key reason **for** the last action:  
6281 上次操作的关键原因:  
6282  
6283  
6284 <\\$key\_reason\_of\_last\_action\\$>  
6285  
6286  
6287  
6288 History Summarization  
6289 历史总结  
6290  
6291  
6292 <\\$history\_summary\\$>  
6293  
6294  
6295  
6296 Success\_Detection flag **for** the overall task:  
6297 整体任务的成功检测标志:  
6298  
6299  
6300 <\\$success\_detection\\$>  
6301  
6302  
6303  
6304 Valid action **set in** Python **format** to select the next action:

6305 用于选择下一步操作的Python格式有效操作集:  
6306  
6307  
6308 <\\$skill\_library\\$>  
6309  
6310  
6311  
6312 Current **and** previous screenshot are the same:  
6313 当前截图与前一截图相同:  
6314  
6315  
6316 <\\$image\_same\_flag\\$>  
6317  
6318  
6319  
6320 Mouse position **in** the current screenshot **is** the same **as** **in** the  
6321 当前截图中的鼠标位置与  
6322  
6323  
6324 previous screenshot:  
6325 前一截图中的相同:  
6326  
6327  
6328 <\\$mouse\_position\_same\_flag\\$>  
6329  
6330  
6331  
6332 Self\_Reflection\_Reasoning:  
6333 自我反思推理:  
6334  
6335  
6336 You need to answer the following questions, step by step, to describe  
6337 你需要逐步回答以下问题, 描述  
6338  
6339  
6340 your reasoning based on the history summarization, last action **and**  
6341 基于历史总结、上次操作及  
6342  
6343  
6344 sequential screenshots of the application during the execution of  
6345 执行上次操作期间应用程序的连续截图,  
6346  
6347  
6348 the last action.  
6349 你的推理过程。  
6350  
6351  
6352 1. Please describe what the page **is in** the current screenshot. Respond  
6353 1. 请描述当前截图中的页面内容。请用一句话回答。  
6354  
6355  
6356 **in** one sentence.  
6357  
6358  
6359  
6360 2. What **is** the last executed action based on the text information

6361 2. 根据上述文本信息，最后执行的操作是什么？

6362  
6363  
6364 above?  
6365  
6366  
6367

6368 3. Was the last executed action successful? Give reasons. You should  
6369 3. 最后执行的操作是否成功？请给出理由。你应当

6370  
6371  
6372 refer to the following rules:  
6373 参考以下规则：

6374  
6375  
6376 - If the action involves moving the mouse, it is considered  
6377 - 如果操作涉及移动鼠标，当鼠标位置在连续截图中保持不变或移动方向错误时，视为不成功，

6378  
6379 unsuccessful when the mouse position remains unchanged or moves in  
6380 无论背景元素和其他项目如何。

6381  
6382  
6383  
6384 an incorrect way across sequential screenshots, regardless of  
6385  
6386  
6387 background elements and other items.

6388  
6389  
6390  
6391 - If the last action executed was empty, then the previous action is  
6392 - 如果最后执行的操作为空，则视前一个操作为成功。

6393  
6394  
6395  
6396 deemed successful.  
6397  
6398  
6399  
6400 - If the last action was related to choose panel, pay attention to the  
6401 - 如果最后的操作与选择面板有关，请注意你所在的面板。该面板是否为你的目标面板？  
6402  
6403  
6404 panel you are in. Does the panel is your target panel?

6405  
6406  
6407  
6408 - If the last action was to drag an element onto the timeline, pay  
6409 - 如果最后的操作是将元素拖动到时间轴上，请注意当前时间轴与  
6410  
6411  
6412  
6413  
6414  
6415  
6416  
---

attention to the difference between the current timeline and the  
之前时间轴的区别。

6417  
6418  
6419  
6420 ---  
6421  
6422  
6423  
6424 previous timeline. Is there the target element you want on the  
6425 前一个时间线。现在时间线上有你想要的目标元素吗?  
6426  
6427  
6428 timeline now?  
6429 时间线上有吗?  
6430  
6431  
6432 - If the last action was related to crop, pay attention to the video  
6433 - 如果上一个操作与裁剪相关, 请注意视频长度。  
6434  
6435  
6436 length. If the video length does **not** change, it **is** considered  
6437 如果视频长度没有变化, 则视为操作失败。  
6438  
6439  
6440 unsuccessful.  
6441 操作失败。  
6442  
6443  
6444 - If the last action executed was '**export\_project()**' and the current  
6445 - 如果上一个执行的操作是"**export\_project()**", 且当前截图是Capcut首页,  
6446  
6447  
6448 screenshot **is** the Capcut homepage, then the previous action **is**  
6449 则认为上一个操作成功。  
6450  
6451  
6452 deemed successful.  
6453 则认为上一个操作成功。  
6454  
6455  
6456 - If the position to move the mouse to was incorrect **and** the mouse  
6457 - 如果鼠标移动的位置不正确, 且鼠标未到达目标UI元素,  
6458  
6459 didn't reach the target UI element, pay more attention to the  
6460 请更加注意准确的移动坐标。  
6461  
6462  
6463  
6464 accurate coordinates to move to.  
6465 请更加注意准确的移动坐标。  
6466  
6467  
6468 - If the action seemed to have no effect, pay attention to the latest  
6469 - 如果操作似乎没有效果, 请关注最新的鼠标位置。  
6470  
6471  
6472 mouse position. Did it move? Did it get closer to the target UI

6473 鼠标移动了吗？是否更接近目标UI元素？  
6474  
6475  
6476 element? Where are the target coordinates **in** the action wrong? The  
6477 操作中的目标坐标哪里出错了?  
6478  
6479  
6480 position of the mouse cursor on the screenshot shows their  
6481 截图上的鼠标光标位置显示了它们的位置。  
6482  
6483  
6484 location.  
6485 截图上的鼠标光标位置显示了它们的位置。  
6486  
6487  
6488 - Was some unrelated UI item triggered by the last action?  
6489 - 上一次操作是否触发了某个无关的界面元素?  
6490  
6491  
6492 4. If the last action **is not** executed successfully, what **is** the most  
6493 4. 如果上一次操作未成功执行，最可能的原因是什么?  
6494  
6495  
6496 probable cause? You should give only one cause **and** refer to the  
6497 你应只给出一个原因，并参照以下规则：  
6498  
6499  
6500 following rules:  
6501  
6502  
6503  
6504 - The reasoning **for** the last action could be wrong.  
6505 - 上一次操作的推理可能是错误的。  
6506  
6507  
6508 - If it was an action involving moving the mouse **or** the text cursor,  
6509 - 如果操作涉及移动鼠标或文本光标，  
6510  
6511  
6512 the most probable cause was that the coordinates used were  
6513 最可能的原因是所用坐标不正确。  
6514  
6515  
6516 incorrect.  
6517  
6518  
6519  
6520 - If it **is** an interaction action, the most probable cause was that the  
6521 - 如果是交互操作，最可能的原因是该操作在当前状态下不可用或未激活。  
6522  
6523  
6524 action was unavailable **or** not activated **in** the current state.  
6525  
6526  
6527  
6528 - If an unrelated change happened **in** the UI, the most probable cause

6529 - 如果界面发生了无关的变化，最可能的原因是操作触发了错误的界面元素。  
6530  
6531  
6532       was that the action triggered an incorrect UI element.  
6533  
6534  
6535  
6536 - If there **is** an error report, analyze the cause based on the report.  
6537 - 如果有错误报告，请根据报告分析原因。  
6538  
6539  
6540 Success\_Detection:  
6541 成功检测：  
6542  
6543  
6544 Based on the history summarization, the last action, the current  
6545 基于历史总结、上一次操作、当前截图和成功检测 (Success\_Detection) 标志，判断是否  
6546  
6547  
6548       screenshots **and** the Success\_Detection flag, determine whether the  
6549  
6550  
6551  
6552       overall task "<\\$task\_description\\$>" was successful. This  
6553 整体任务"<\\$task\_description\\$>"已成功完成。这  
6554  
6555  
6556       assessment should consider the overall task's success, not just  
6557 评估应考虑整体任务的成功，而不仅仅是  
6558  
6559  
6560       individual actions.  
6561 单个操作。  
6562  
6563  
6564 - If the last action executed was an empty **list** and "<  
6565 - 如果最后执行的操作是空列表且"<  
6566  
6567  
6568       \\$success\_detection\\$>" indicates the task is successful, then the  
6569 \\$success\_detection\\$>"表明任务成功，则  
6570  
6571  
6572       overall task has a high chance of being considered a success.  
6573 整体任务很可能被视为成功。  
6574  
6575  
6576 - If the overall task was unsuccessful, specify the reason of failure  
6577 - 如果整体任务未成功，请说明失败原因  
6578  
6579  
6580       **and** which steps are missing.  
6581 及缺失的步骤。  
6582  
6583  
6584 - If the overall task was successful, ONLY output "SUCCESSFUL".

6585 - 如果整体任务成功，仅输出 "SUCCESSFUL"。  
6586  
6587  
6588 You should only respond **in** the format as described below.  
6589 你应仅以以下描述的格式回复。  
6590  
6591  
6592 Self\_Reflection\_Reasoning:  
6593 自我反思推理：  
6594  
6595  
6596 1. ...  
6597  
6598  
6599  
6600 2. ...  
6601  
6602  
6603  
6604 3. ...  
6605  
6606  
6607  
6608 Success\_Detection:  
6609 成功检测：  
6610  
6611  
6612 ...  
6613  
6614  
6615  
6616 ---  
6617  
6618  
6619  
6620 Prompt 36: Capcut: Task Inference prompt.  
6621 提示36：Capcut：任务推断提示。  
6622  
6623  
6624 ---  
6625  
6626  
6627  
6628 Assume you are a helpful AI assistant integrated **with 'CapCut'** on the  
6629 假设你是集成在PC端"CapCut"中的智能助手，  
6630  
6631  
6632 the PC, equipped to handle a wide **range** of tasks **in** the game.  
6633 能够处理游戏中的各种任务。  
6634  
6635  
6636 Capcut **is** a video editing software. You will be sequentially given  
6637 Capcut是一款视频编辑软件。你将依次接收  
6638  
6639  
6640 <\\$event\_count\\$> screenshots **and** corresponding descriptions of

6641 <\$event\_count\$> 张截图及其对应的近期事件描述。  
6642  
6643  
6644 recent events. You will also be given a summary of the history  
6645 还将提供最后一张截图之前发生的历史摘要。  
6646  
6647  
6648 that happened before the last screenshot. You should assist in  
6649 您应协助总结事件以便未来决策，  
6650  
6651  
6652 summarizing the events for future decision-making and also in  
6653 并提出在目标任务下最合适执行的子任务。  
6654  
6655  
6656 proposing the most suitable subtask to execute next, given the  
6657 给定目标任务，建议最适合执行的子任务。  
6658  
6659  
6660 target task.  
6661 目标任务。  
6662  
6663  
6664 ---  
6665  
6666  
6667  
6668 ---  
6669  
6670  
6671  
6672 Here is some helpful information to help you do the summarization and  
6673 以下是帮助您进行总结和提出子任务的一些有用信息。  
6674  
6675  
6676 propose the subtask.  
6677 帮助您进行总结和提出子任务。  
6678  
6679  
6680 Overall task description:  
6681 整体任务描述：  
6682  
6683  
6684 <\\$task\_description\\$>  
6685  
6686  
6687  
6688 Previous proposed subtask for the task:  
6689 之前为该任务提出的子任务：  
6690  
6691  
6692 <\\$subtask\_description\\$>  
6693  
6694  
6695  
6696 Previous reasoning for proposing the subtask:

6697 提出孩子任务的前期推理:  
6698  
6699  
6700 <\\$subtask\_reasoning\\$>  
6701  
6702  
6703  
6704 Image introduction:  
6705 图片介绍:  
6706  
6707  
6708 <\\$image\_introduction\\$>  
6709  
6710  
6711  
6712 Last executed action:  
6713 最后执行的操作:  
6714  
6715  
6716 <\\$previous\_action\\$>  
6717  
6718  
6719  
6720 Error report for the last executed action:  
6721 最后执行操作的错误报告:  
6722  
6723  
6724 <\\$executing\_action\_error\\$>  
6725  
6726  
6727  
6728 key decision-making reasoning for the last executed action:  
6729 最后执行操作的关键决策推理:  
6730  
6731  
6732 <\\$previous\_reasoning\\$>  
6733  
6734  
6735  
6736 Self-reflection for the last executed action:  
6737 最后执行操作的自我反思:  
6738  
6739  
6740 <\\$self\_reflection\_reasoning\\$>  
6741  
6742  
6743  
6744 Success\_Detection for the overall task:  
6745 整体任务的成功检测:  
6746  
6747  
6748 <\\$success\_detection\\$>  
6749  
6750  
6751  
6752 The following is the summary of history that happened before the last

6753 以下是发生在上一次之前的历史总结  
6754  
6755  
6756 screenshot:  
6757 截图:  
6758  
6759  
6760 <\\$previous\_summarization\\$>  
6761  
6762  
6763  
6764 History\_summary: Summarize what happened **in** the past experience,  
6765 历史总结：总结过去的经历发生了什么，  
6766  
6767  
6768 especially the last step according to the decision-making  
6769 尤其是根据决策的最后一步  
6770  
6771  
6772 reasoning **and** self-reflection reasoning **for** the last executed  
6773 最近执行的推理与自我反思推理  
6774  
6775  
6776 action. The summarization needs to be precise, concrete, highly  
6777 行动。总结需要精准、具体且高度  
6778  
6779  
6780 related to the task, **and** follow the rules below.  
6781 与任务相关，并遵守以下规则。  
6782  
6783  
6784 1. Determine **if** the task has been completed successfully. If it **is**  
6785 1. 确定任务是否已成功完成。如果是  
6786  
6787  
6788 successful, ignore question **2** to **5**.  
6789 成功，忽略问题2到5。  
6790  
6791  
6792 2. Summarize the tasks **from** the history **and** the current task. What **is**  
6793 2. 总结历史任务和当前任务。内容是什么  
6794  
6795  
6796 the current progress of the task? For example, to **open** a file, you  
6797 任务的当前进展？例如，要打开一个文件，你  
6798  
6799  
6800 first need to select the file, then **open** it by clicking somewhere  
6801 首先需要选择文件，然后点击某处打开它  
6802  
6803  
6804 or using the keyboard. Subtasks may have other pre-requisites.  
6805 或使用键盘。子任务可能有其他先决条件。  
6806  
6807  
6808 3. Record the successful actions **and** organize them into events, step

6809 3. 记录成功的操作并将其组织成事件，步骤  
6810  
6811  
6812       by step.  
6813       逐步进行。  
6814  
6815  
6816 4. Which subtask has been completed? Which subtasks have not? Do not  
6817 4. 哪个子任务已完成？哪些子任务尚未完成？不要  
6818  
6819  
6820       forget the information and key events in the previous steps of the  
6821       忘记之前步骤中的信息和关键事件  
6822  
6823  
6824       overall task.  
6825       整体任务。  
6826  
6827  
6828 Subtask\_reasoning: Decide whether the previous subtask is finished and  
6829 子任务推理：判断前一个子任务是否已完成并  
6830  
6831  
6832       whether it is necessary to propose a new subtask. The subtask  
6833       是否有必要提出一个新的子任务。该子任务  
6834  
6835  
6836       should be straightforward, contribute to the target task, and be  
6837       应该是直接明了的，有助于目标任务，并且  
6838  
6839  
6840       most suitable for the current situation; which should be completed  
6841       最适合当前情况；应当完成  
6842  
6843  
6844       within a few actions. You should respond with:  
6845       在几步操作内。你应该回复：  
6846  
6847  
6848 1. How to finish the target task? You should analyze it step by step.  
6849 1. 如何完成目标任务？你应该逐步分析。  
6850  
6851  
6852 - To add Media, Audio, Text, Stickers, Effects, Transitions, Filters,  
6853 - 添加媒体、音频、文本、贴纸、特效、转场、滤镜，  
6854  
6855  
6856       Adjustments or Templates, you should first switch to that panel  
6857       调整或模板，您应先切换到该面板  
6858  
6859  
6860       and then drag the target object to the video in the timeline.  
6861       然后将目标对象拖动到时间线中的视频上。  
6862  
6863  
6864 - To get content information of a video, you can use related skills.

6865 - 要获取视频的内容信息，您可以使用相关技能。  
6866  
6867  
6868 For example, you want to know which exactly second you want to  
6869 例如，你想知道确切的哪一秒你想要  
6870  
6871  
6872 operate.  
6873 操作。  
6874  
6875  
6876 2. What **is** the current progress of the target task according to the  
6877 2. 根据当前情况，目标任务的进展如何  
6878  
6879  
6880 analysis **in** question 1? Please do **not** make **any** assumptions **if** they  
6881 问题1中的分析？如果没有明确说明，请不要做任何假设  
6882  
6883  
6884 ---  
6885  
6886  
6887  
6888 ---  
6889  
6890  
6891  
6892 are **not** mentioned **in** the above information. You should assume  
6893 未在上述信息中提及。您应当假设  
6894  
6895  
6896 that you are doing the task **from** scratch. Please strictly follow  
6897 你是从零开始执行该任务的。请严格遵守  
6898  
6899  
6900 the description **and** requirements **in** the current task.  
6901 当前任务中的描述和要求。  
6902  
6903  
6904 3. What **is** the previous subtask? Has the previous subtask finished due  
6905 3. 上一个子任务是什么？上一个子任务是否已完成  
6906  
6907  
6908 to **self**-reflection? Or **is** it improper **for** the current situation?  
6909 进行自我反思？还是说这对当前情况不合适？  
6910  
6911  
6912 If finished **or** improper, please select a new one, otherwise you  
6913 如果已完成或不合适，请选择一个新的，否则您  
6914  
6915  
6916 should reuse the last subtask.  
6917 应重复使用最后一个子任务。  
6918  
6919  
6920 4. If you want to propose a new subtask, give reasons why it **is** more

6921 4. 如果您想提出一个新的子任务, 请说明其更为合理的原因  
6922  
6923  
6924     feasible **for** the current situation. Please strictly follow the  
6925     适合当前情况。请严格遵守  
6926  
6927  
6928     description **and** requirements **in** the current task.  
6929     当前任务中的描述和要求。  
6930  
6931  
6932 5. The proposed subtask needs to be precise **and** concrete within one  
6933 5. 所提出的子任务需要在一个范围内明确且具体  
6934  
6935  
6936     sentence. It should **not** be directly related to **any** skills.  
6937     句子。它不应直接与任何技能相关。  
6938  
6939  
6940 You should only respond **in** the **format** described below, **and** you should  
6941 您应仅以以下描述的格式回复, 并且您应  
6942  
6943  
6944     **not** output comments **or** other information.  
6945     不输出评论或其他信息。  
6946  
6947  
6948 History\_summary:  
6949 历史概述:  
6950  
6951  
6952 1. ...  
6953  
6954  
6955  
6956 2. ...  
6957  
6958  
6959  
6960 ...  
6961  
6962  
6963  
6964 Subtask\_reasoning:  
6965 子任务推理:  
6966  
6967  
6968 1. ...  
6969  
6970  
6971  
6972 2. ...  
6973  
6974  
6975  
6976 ...

6977  
6978  
6979  
6980 Subtask\_description:  
6981 子任务描述:  
6982  
6983  
6984 The current subtask is ...  
6985 当前子任务是 ...  
6986  
6987  
6988 ----  
6989  
6990  
6991  
6992 Prompt 37: Capcut: Screen Classification prompt.  
6993 提示37: Capcut: 屏幕分类提示。  
6994  
6995  
6996 ----  
6997  
6998  
6999  
7000 You are an assistant who assesses my progress in playing Red Dead  
7001 你是一个评估我玩《荒野大镖客》(Red Dead) 进展的助手  
7002  
7003  
7004 Redemption 2 on the PC and provides expert guidance. Imagine you  
7005 《荒野大镖客救赎2》电脑版并提供专业指导。想象一下你  
7006  
7007  
7008 are playing Red Dead Redemption 2 with the keyboard and mouse, the  
7009 正在使用键盘和鼠标玩《荒野大镖客救赎2》  
7010  
7011  
7012 image is the screenshot of your computer.  
7013 图像是您电脑的截图。  
7014  
7015  
7016 Given the classes, please select the class that best describes the  
7017 鉴于这些类别, 请选择最能描述的类别  
7018  
7019 screenshot.  
7020 截图。  
7021  
7022  
7023  
7024 <classes>  
7025  
7026  
7027  
7028 You must follow the following criteria:  
7029 您必须遵守以下标准:  
7030  
7031  
7032 (1) The output should only be a JSON file. You should not add any

7033 (1) 输出应仅为 JSON 文件。您不应添加任何内容  
7034  
7035  
7036 other explanation text along with the JSON.  
7037 其他与 JSON 一起的说明文本。  
7038  
7039  
7040 (2) You should choose one class for the value of "class".  
7041 (2) 你应为 "class" 的值选择一个类别。  
7042  
7043  
7044 (3) Do not change the "type": "screen\_classification" in your output.  
7045 (3) 不要更改输出中的 "type": "screen\_classification"。  
7046  
7047  
7048 The output format should be as follows:  
7049 输出格式应如下所示:  
7050  
7051  
7052 Classes:  
7053 类别:  
7054  
7055  
7056 map  
7057 地图  
7058  
7059  
7060 ---  
7061  
7062  
7063  
7064 Prompt 38: Capcut: Action Planning prompt.  
7065 提示38: Capcut: 操作规划提示。  
7066  
7067  
7068 ---  
7069  
7070  
7071  
7072 You are a helpful AI assistant integrated with 'CapCut' on the PC,  
7073 你是集成在PC端 "CapCut" 中的智能助手,  
7074  
7075 equipped to handle a wide range of tasks in the application.  
7077 能够处理应用中的各种任务。  
7078  
7079  
7080 Capcut is a video editing software. Your advanced capabilities  
7081 Capcut 是一款视频编辑软件。你的高级功能  
7082  
7083 enable you to process and interpret application screenshots and  
7085 使你能够处理和解读应用截图及  
7086  
7087  
7088 other relevant information. By analyzing these inputs, you gain a

7089      其他相关信息。通过分析这些输入，你获得了  
7090  
7091  
7092      comprehensive understanding of the current context **and** situation  
7093      对应用当前上下文和情境的全面理解。  
7094  
7095  
7096      within the application. Utilizing these insights, you are tasked  
7097      利用这些洞察，你的任务是  
7098  
7099  
7100      **with** identifying the most suitable **in-application** action to take  
7101      在当前任务下，识别最合适的应用内操作步骤，  
7102  
7103  
7104      **next**, given the current task. You control the application **and** can  
7105      你可以控制应用并执行可用操作集中的动作，  
7106  
7107  
7108      execute actions **from** the available action **set** to manipulate its UI  
7109      以操控其用户界面。  
7110  
7111  
7112      . Upon evaluating the provided information, your role **is** to  
7113      在评估所提供的信息后，你的职责是  
7114  
7115  
7116      articulate the precise actions you should perform, considering the  
7117      准确表达你应该执行的具体操作，考虑  
7118  
7119  
7120      ---  
7121  
7122  
7123  
7124      ---  
7125  
7126  
7127  
7128      application's present circumstances, and specify any necessary  
7129      应用当前的状况，并说明任何必要的  
7130  
7131  
7132      parameters **for** implementing that action.  
7133      执行该操作的参数。  
7134  
7135  
7136      Here **is** some helpful information to **help** you make the decision.  
7137      这里有一些有助于您做出决定的信息。  
7138  
7139  
7140      Overall task description:  
7141      整体任务描述：  
7142  
7143  
7144      <\\$task\_description\\$>

7145  
7146  
7147  
7148 Subtask description:  
7149 子任务描述:  
7150  
7151 <\\$subtask\_description\\$>  
7153  
7154  
7155  
7156 Few shots:  
7157 少量示例:  
7158  
7159  
7160 <\\$few\_shots\\$>  
7161  
7162  
7163  
7164 Image introduction:  
7165 图片介绍:  
7166  
7167  
7168 <\\$image\_introduction\\$>  
7169  
7170  
7171  
7172 Current **and** previous screenshot are the same:  
7173 当前截图与之前截图相同:  
7174  
7175  
7176 <\\$image\_same\_flag\\$>  
7177  
7178  
7179  
7180 Mouse position **in** the current screenshot **is** the same **as** **in** the  
7181 当前截图中的鼠标位置与  
7182  
7183  
7184 previous screenshot:  
7185 之前截图中的相同:  
7186  
7187  
7188 <\\$mouse\_position\_same\_flag\\$>  
7189  
7190  
7191  
7192 Description of current screenshot:  
7193 当前截图描述:  
7194  
7195  
7196 <\\$image\_description\\$>  
7197  
7198  
7199  
7200 Description of label IDs:

7201 标签ID描述:  
7202  
7203  
7204 <\\$description\_of\_bounding\_boxes\\$>  
7205  
7206  
7207  
7208 Last executed action:  
7209 上次执行的操作:  
7210  
7211  
7212 <\\$previous\_action\\$>  
7213  
7214  
7215  
7216 Key reason for the last action:  
7217 上次操作的关键原因:  
7218  
7219  
7220 <\\$key\_reason\_of\_last\_action\\$>  
7221  
7222  
7223  
7224 Self-reflection for the last executed action:  
7225 对上次执行操作的自我反思:  
7226  
7227  
7228 <\\$previous\_self\_reflection\_reasoning\\$>  
7229  
7230  
7231  
7232 Summarization of recent history:  
7233 近期历史总结:  
7234  
7235  
7236 <\\$previous\_summarization\\$>  
7237  
7238  
7239  
7240 Valid action set in Python format to select the next action:  
7241 用于选择下一步操作的Python格式有效操作集:  
7242  
7243  
7244 <\\$skill\_library\\$>  
7245  
7246  
7247  
7248 Success\_Detection for overall task:  
7249 整体任务的成功检测:  
7250  
7251  
7252 <\\$success\_detection\\$>  
7253  
7254  
7255  
7256 Based on the above information, you should first analyze the current

7257 基于以上信息，您应首先分析当前  
7258  
7259  
7260 situation and provide the reasoning for what you should do for the  
7261 情况，并说明完成任务下一步应采取的理由。  
7262  
7263  
7264 next step to complete the task. Then, you should output the exact  
7265 然后，您应输出在应用程序中要执行的具体  
7266  
7267  
7268 action you want to execute in the application.  
7269 操作。  
7270  
7271  
7272 Pay attention to all UI items and contents in the image. DO NOT make  
7273 注意图像中的所有界面元素和内容。不要对布局做出  
7274  
7275  
7276 assumptions about the layout! If the image includes a mouse cursor  
7277 假设！如果图像中包含鼠标光标，  
7278  
7279  
7280 , pay close attention to the coordinates of the pointer tip, not  
7281 请特别关注指针尖端的坐标，而非  
7282  
7283  
7284 the centre of the mouse cursor.  
7285 鼠标光标的中心。  
7286  
7287  
7288 You should respond to me with the following information, and you MUST  
7289 您应按顺序向我反馈以下信息，且必须  
7290  
7291  
7292 respond one by one.  
7293 逐条回应。  
7294  
7295  
7296 Decision\_Making\_Reasoning: You should think step by step and provide  
7297 决策推理：您应逐步思考并提供  
7298  
7299  
7300 detailed reasoning to determine the next action executed on the  
7301 详细推理，以确定在当前任务状态下执行的下一步操作。  
7302  
7303  
7304 current state of the task.  
7305  
7306  
7307  
7308 1. Does "<\$success\_detection\$>" means the overall task was  
7309 1. "<\$success\_detection\$>"是否表示整体任务  
7310  
7311  
7312 successful? If successful, ignore questions 2-11.

7313 成功？如果成功，则忽略第2至11题。  
7314  
7315  
7316 ---  
7317  
7318  
7319  
7320 ---  
7321  
7322  
7323  
7324 2. Which skill **in** the Skill Library "<\\$skill\_library\\$>" has the  
7325 2. 在技能库"<\\$skill\_library\\$>"中，哪个技能具有  
7326  
7327  
7328 closest semantics to the current subtask "<\\$subtask\_description\\$"  
7329 与当前子任务"<\\$subtask\_description\\$>"最接近的语义  
7330  
7331  
7332 >"? If there is an answer, select it as the output action.  
7333 >"? 如果有答案，请选择它作为输出操作。  
7334  
7335  
7336 3. Prefer keyboard operation over mouse operation. Is there a  
7337 3. 优先使用键盘操作而非鼠标操作。是否存在  
7338  
7339  
7340 direct skill **in** the skill library to complete the current action?  
7341 直接使用技能库中的技能来完成当前动作吗？  
7342  
7343  
7344 If there **is**, please specify which it **is**. Or are there **any** keyboard  
7345 如果有，请具体说明是哪一个。或者是否有任何键盘  
7346  
7347  
7348 actions, such **as** using shortcut keys **or** pressing "enter", to  
7349 操作，例如使用快捷键或按"回车"，以  
7350  
7351  
7352 finish current step **or** overall task? Please specify which it **is**.  
7353 完成当前步骤还是整个任务？请具体说明。  
7354  
7355  
7356 4. Always **try** pressing "enter" first instead of clicking it **with**  
7357 4. 总是先尝试按"回车"键，而不是用鼠标点击它  
7358  
7359  
7360 the mouse, **if** the button you want to click **is** active.  
7361 鼠标，如果你想点击的按钮是激活状态。  
7362  
7363  
7364 5. If you need to get information **from** video content, select the  
7365 5. 如果您需要从视频内容中获取信息，请选择  
7366  
7367  
7368 skill **get\_information\_from\_video()**. For example, you want to know

7369 技能 `get_information_from_video()`。例如，你想知道  
7370  
7371  
7372 which exactly second you want to operate.  
7373 你想操作的确切秒数。  
7374  
7375  
7376 6. Based on the current screenshot **and** the description of label  
7377 6. 基于当前截图和标签描述  
7378  
7379  
7380 IDs **in** text, which label ID **is** most relevant to the current task?  
7381 文本中的ID，哪个标签ID与当前任务最相关？  
7382  
7383  
7384 You should never answer this question based on the screenshot.  
7385 你绝不应该仅凭截图来回答这个问题。  
7386  
7387  
7388 7. If the previous action **is** unsuccessful, DO NOT repeat the  
7389 7. 如果前一个操作失败，请勿重复执行  
7390  
7391  
7392 previous action, consider an alternative action **if** possible. Such  
7393 先前的操作，如可能，考虑替代操作。此类  
7394  
7395  
7396 as click different label ID **or** use different shortcut keys. If  
7397 通过点击不同的标签ID或使用不同的快捷键。如果  
7398  
7399  
7400 there **is** an alternative action, please specify what it **is**.  
7401 如果有其他可选操作，请说明具体内容。  
7402  
7403  
7404 8. In the current screenshot, identify the label ID of the  
7405 8. 在当前截图中，识别该标签的ID  
7406  
7407  
7408 bounding box most relevant to the current step. If there **is** text  
7409 与当前步骤最相关的边界框。如果有文本  
7410  
7411  
7412 within this bounding box, please provide the text.  
7413 请在此边界框内输入文本。  
7414  
7415  
7416 9. If mouse actions are necessary, use that specify bounding box  
7417 9. 如果需要鼠标操作，请使用指定的边界框  
7418  
7419  
7420 label ID (**if** shown **in** the current screenshot) **as** parameter, rather  
7421 将标签ID（如果在当前截图中显示）作为参数，而不是  
7422  
7423  
7424 than directly generating normalized \$x\$ **and** \$y\$ coordinates. If there

7425 比直接生成归一化的\$x\$和\$y\$坐标更好。如果有  
7426  
7427  
7428 **is any** relevant label ID, please specify which it **is**.  
7429 如果是任何相关的标签ID, 请具体说明是哪一个。  
7430  
7431  
7432 10. If there **is** a dialog **open** after the previous action, pay  
7433 10. 如果前一个操作后有对话框打开, 请注意  
7434  
7435  
7436 attention to **any** missing step before clicking on it's **buttons**. For  
7437 点击其按钮前请注意是否有遗漏的步骤。  
7438  
7439  
7440 example, before clicking "Save", make sure the file name **is** typed  
7441 例如, 点击"保存"之前, 请确保已输入文件名  
7442  
7443  
7444 in the correct text field.  
7445 在正确的文本字段中。  
7446  
7447  
7448 11. If you need to use an action outside an **open** menu **or** dialog,  
7449 11. 如果需要在打开的菜单或对话框之外使用某个操作,  
7450  
7451  
7452 please close the current menu **or** dialog before trying the **next**  
7453 请先关闭当前菜单或对话框, 然后再尝试下一步  
7454  
7455  
7456 action.  
7457 行动。  
7458  
7459  
7460 Actions: The best action, **or** short sequence of actions without gaps,  
7461 操作: 最佳操作, 或无间断的短操作序列,  
7462  
7463  
7464 to execute **next** to progress **in** achieving the goal. Pay attention  
7465 执行下一步以推进目标的实现。请注意  
7466  
7467  
7468 to the names of the available skills **and** the previous skills  
7469 到可用技能的名称和之前的技能  
7470  
7471  
7472 already executed, **if any**. Pay special attention to the coordinates  
7473 已执行 (如有)。特别注意坐标  
7474  
7475  
7476 of **any** action that needs them. Do **not** make assumptions about the  
7477 任何需要它们的操作。不要对其做出假设  
7478  
7479  
7480 location of UI elements **or** their coordinates, analyse **in** detail

7481 界面元素的位置或其坐标，进行详细分析  
7482  
7483  
7484 any provided images. You should also pay more attention to the  
7485 任何提供的图像。你还应更加注意  
7486  
7487  
7488 following action rules:  
7489 以下操作规则：  
7490  
7491  
7492 1. If "<\\$success\_detection\\$>" means the overall task was  
7493 1. 如果"<\\$success\_detection\\$>"表示整体任务已  
7494  
7495  
7496 successful or equal to "True", then output action MUST be empty  
7497 成功或等于"True"，则输出操作必须为空  
7498  
7499  
7500 like ". Be careful to check the task was really successful.  
7501 像"。请务必确认任务确实成功完成。  
7502  
7503  
7504 2. You should output actions in Python code format and specify any  
7505 2. 你应以Python代码格式输出操作，并说明任何  
7506  
7507  
7508 necessary parameters to execute that action. Only use function  
7509 执行该操作所需的参数。仅使用函数  
7510  
7511  
7512 names and argument names exactly as shown in the valid actions et.  
7513 名称和参数名称与有效操作集中的完全一致。  
7514  
7515  
7516 If a function has parameters, you should also include their names  
7517 如果函数有参数，也应包含它们的名称  
7518  
7519  
7520 and decide their values, like "press\_shift (duration=1)". If it  
7521 并决定它们的值，例如"press\_shift (持续时间=1)"。如果它  
7522  
7523  
7524 does not have a parameter, just output the action, like "  
7525 没有参数，只需输出动作，如"  
7526  
7527  
7528 release\_mouse\_buttons().  
7529 释放鼠标按钮()  
7530  
7531  
7532 4. Given the current situation and task, you should only choose  
7533 鉴于当前情况和任务，你只应选择  
7534  
7535  
7536 the most suitable action from the valid action set. You cannot use

7537 从有效动作集中选择最合适的作品。你不能使用  
7538  
7539  
7540 actions that are **not in** the valid action **set** to control the  
7541 不在有效动作集合中的动作用于控制  
7542  
7543  
7544 application.  
7545 应用程序。  
7546  
7547  
7548 5. When you decide to perform a mouse action, **if** there **is** bounding  
7549 5. 当您决定执行鼠标操作时，如果存在边界  
7550  
7551  
7552 box **in** the current screenshot, you **MUST** choose skill  
7553 在当前截图中的框内，您必须选择技能  
7554  
7555  
7556 click\_on\_label(label\_id, mouse\_button).  
7557 点击标签(label\_id, 鼠标按钮)。  
7558  
7559  
7560 6. When you perform a mouse action, always **select** the target UI  
7561 6. 当你执行鼠标操作时，总是选择目标用户界面  
7562  
7563  
7564 element closest to the UI element of the previous action **for**  
7565 最接近前一个操作的UI元素的元素，用于  
7566  
7567  
7568 operation.  
7569 操作。  
7570  
7571  
7572 7. When you decide to perform a mouse click, prioritize clicking  
7573 7. 当你决定执行鼠标点击时，优先点击  
7574  
7575  
7576 icons, instead of text.  
7577 图标，而非文字。  
7578  
7579  
7580 ---  
7581  
7582  
7583  
7584 ---  
7585  
7586  
7587  
7588 8. When there **is** new dialog box that affects the **next** step, you  
7589 8. 当出现影响下一步的新对话框时，您  
7590  
7591  
7592 should close it.

7593 应该关闭它。  
7594  
7595  
7596 9. The material panel includes the Media, Audio, Text, Stickers,  
7597 9. 材料面板包括媒体、音频、文本、贴纸。  
7598  
7599  
7600 Effects, Transitions, Filters, Adjustments, **and** Templates tabs.  
7601 效果、转场、滤镜、调整和模板标签。  
7602  
7603  
7604 Choose this skill "switch\_material\_panel()" to switch between  
7605 选择此技能 "switch\_material\_panel()" 以切换  
7606  
7607  
7608 these tabs one by one.  
7609 逐个浏览这些标签。  
7610  
7611  
7612 10. To add media, drag that media to the video **in** the timeline.  
7613 10. 要添加媒体，请将该媒体拖动到时间线中的视频上。  
7614  
7615  
7616 Key\_reason\_of\_last\_action: Summarize the key reasons why you output  
7617 上一次操作的主要原因：总结你输出的关键原因  
7618  
7619  
7620 this action.  
7621 此操作。  
7622  
7623  
7624 You should only respond **in** the **format** described below. In your  
7625 您应仅以以下描述的格式进行回复。在您的  
7626  
7627 reasoning **for** the chosen actions, also describe which item you  
7629 所选行动的推理，同时描述你选择了哪个项目  
7630  
7631  
7632 decided to interact **with** **and** why. DO NOT change the title of each  
7633 决定与谁互动以及原因。不要更改每个标题  
7634  
7635  
7636 item. You should **not** output other comments **or** information besides  
7637 项目。您不应输出除外的其他评论或信息  
7638  
7639  
7640 the **format** below.  
7641 以下格式。  
7642  
7643  
7644 Decision\_Making\_Reasoning:  
7645 决策推理：  
7646  
7647  
7648 1. ...

7649  
7650  
7651  
7652 2. ...  
7653  
7654  
7655  
7656 3. ...  
7657  
7658  
7659  
7660 ...  
7661  
7662  
7663  
7664 Actions:  
7665 操作:  
7666  
7667  
7668 44 CPUTH ON  
7669 44 CPUTH 开启  
7670  
7671  
7672 action(args1=x,args2=y)  
7673 action(args1=x,args2=y)  
7674  
7675  
7676 ccc  
7677 ccc  
7678  
7679  
7680 Key\_reason\_of\_last\_action:  
7681 上次操作的关键原因:  
7682  
7683  
7684 ...  
7685  
7686  
7687  
7688 ---  
7689  
7690  
7691  
7692 Prompt 39: Meitu: Information Gathering prompt.  
7693 提示39: 美图: 信息收集提示。  
7694  
7695  
7696 ---  
7697  
7698  
7699  
7700 Assume you are a helpful AI assistant integrated with 'Meitu Xiuxiu'  
7701 假设你是集成在"美图秀秀"(Meitu Xiuxiu)中的智能助手  
7702  
7703  
7704 on the PC, equipped to handle a wide range of tasks in the

7705 在PC端，能够处理应用中的各种任务。  
7706  
7707  
7708 application. Meitu Xiuxiu **is** a user-friendly **and** powerful image  
7709 美图秀秀是一款用户友好且功能强大的图像  
7710  
7711  
7712 editing **and** beautification software. Your advanced capabilities  
7713 编辑与美化软件。你具备先进的能力，  
7714  
7715  
7716 enable you to process **and** interpret application screenshots **and**  
7717 能够处理和解析应用截图及  
7718  
7719  
7720 other relevant information.  
7721 其他相关信息。  
7722  
7723  
7724 Image introduction:  
7725 图片介绍:  
7726  
7727  
7728 <\\$image\_introduction\\$>  
7729  
7730  
7731  
7732 Overall task:  
7733 整体任务:  
7734  
7735  
7736 <\\$task\_description\\$>  
7737  
7738  
7739  
7740 Subtask description:  
7741 子任务描述:  
7742  
7743  
7744 <\\$subtask\_description\\$>  
7745  
7746  
7747  
7748 Image\_Description:  
7749 图片描述:  
7750  
7751 1. Please describe the screenshot image **in** detail. Pay attention to  
7752 1. 请详细描述截图图片。注意  
7753  
7754  
7755  
7756 any details **in** the image, **if any**, especially critical icons, **or**  
7757 图像中的任何细节（如果有），尤其是关键图标，或  
7758  
7759  
7760 created items.

7761 已创建的项目。  
7762  
7763  
7764 2. If the image includes a mouse cursor, please describe what UI  
7765 2. 如果图像中包含鼠标光标, 请描述界面(UI)的内容  
7766  
7767  
7768 element the mouse **is** currently located near. Pay attention to the  
7769 鼠标当前所在的元素附近。注意  
7770  
7771  
7772 coordinates of the pointer tip, **not** the center of the mouse cursor  
7773 指针尖端的坐标, 而非鼠标光标的中心  
7774  
7775  
7776 3. Pay attention to **all** UI items **and** contents **in** the image. Do **not**  
7777 3. 注意图像中的所有用户界面项目和内容。不要  
7778  
7779  
7780 make assumptions about the layout.  
7781 对布局做出假设。  
7782  
7783  
7784 Description\_of\_bounding\_boxes:  
7785 边界框描述:  
7786  
7787  
7788 Please provide a **list** of EVERY bounding box **from** label ID of **1** to <  
7789 请提供标签ID从1到<的所有边界框列表  
7790  
7791  
7792 **\\$length\_of\_som\_map\\$** ONE BY ONE. The label IDs are marked **in** the  
7793 **\\$length\_of\_som\_map\\$** 一一对应。标签ID标记在  
7794  
7795  
7796 upper left corner of the bounding boxes.  
7797 边界框的左上角。  
7798  
7799  
7800 For bounding boxes containing text, provide ONLY the text.  
7801 对于包含文本的边界框, 仅提供文本内容。  
7802  
7803  
7804 For bounding boxes without text, brief description of the function.  
7805 对于无文本的边界框, 简要描述其功能。  
7806  
7807  
7808 ---  
7809  
7810  
7811  
7812 ---  
7813  
7814  
7815  
7816 Format your response **as follows:** ' 1: function\_a', '2: text\_b', ... ,

7817 请按以下格式回复: ' 1: function\_a', '2: text\_b', ... ,  
7818  
7819  
7820     \\\$length\_of\_som\_map\\\$: function\_b'. Don't write anything you are  
7821     \\\$length\_of\_som\_map\\\$: function\_b'。不要写任何你是  
7822  
7823  
7824     not sure about.  
7825     不确定。  
7826  
7827  
7828 Target\_object\_name: Assume you can use an object detection model to  
7829     目标对象名称：假设您可以使用一个目标检测模型来  
7830  
7831  
7832     detect the most relevant object or UI item for completing the  
7833     检测完成任务最相关的对象或界面元素  
7834  
7835  
7836     current task if needed. What item should be detected to complete  
7837     如有需要，完成当前任务应检测哪个项目  
7838  
7839  
7840     the task based on the current screenshot and the current task? You  
7841     基于当前截图和当前任务的任务？你  
7842  
7843  
7844     should obey the following rules:  
7845     应遵守以下规则：  
7846  
7847  
7848 1. Identify an item that is relevant to the current or intermediate  
7849 1. 确定一个与当前或中间阶段相关的项目  
7850  
7851  
7852     target of the task. If the item is within a bounding box in the  
7853     任务的目标。如果该项目位于边界框内  
7854  
7855  
7856     Screenshot, please include the corresponding label ID.  
7857     截图时，请包含相应的标签ID。  
7858  
7859  
7860 2. If no explicit item is specified, only output "null".  
7861 2. 如果未指定明确的项目，则仅输出"null"。  
7862  
7863  
7864 3. If there is no need to detect an object, only output "null".  
7865 3. 如果不需要检测对象，则仅输出"null"。  
7866  
7867  
7868 Reasoning\_of\_object: Why was this object chosen, or why is there no  
7869 对象的推理：为什么选择了这个对象，或者为什么没有选择  
7870  
7871  
7872     need to detect an object?

7873 需要检测一个物体吗?  
7874  
7875  
7876 You should only respond **in the format** described below **and not** output  
7877 您应仅以以下描述的格式回复, 不要输出其他内容  
7878  
7879  
7880 comments **or** other information. DO NOT change the title of each  
7881 评论或其他信息。请勿更改每个标题  
7882  
7883  
7884 item.  
7885 项目。  
7886  
7887  
7888 Image\_Description:  
7889 图像描述:  
7890  
7891  
7892 1. ...  
7893  
7894  
7895  
7896 2. ...  
7897  
7898  
7899  
7900 3. ...  
7901  
7902  
7903  
7904 Description\_of\_bounding\_boxes:  
7905 边界框描述:  
7906  
7907  
7908 Format like: 1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$  
7909 格式如: 1: function\_a', '2: text\_b', ..., '<\\$len\_of\_bound\_boxes\\$  
7910  
7911  
7912 >: function\_b  
7913 >: function\_b  
7914  
7915  
7916 Target\_object\_name:  
7917 目标对象名称:  
7918  
7919  
7920 label ID, Name  
7921 标签ID, 名称  
7922  
7923  
7924 Reasoning\_of\_object:  
7925 对象推理:  
7926  
7927  
7928 ...

7929  
7930  
7931  
7932 ---  
7933  
7934  
7935  
7936 Prompt 40: Meitu: Self Reflection prompt.  
7937 提示40：美图秀秀：自我反思提示。  
7938  
7939  
7940 ---  
7941  
7942  
7943  
7944 Assume you are a helpful AI assistant integrated with 'Meitu Xiuxiu'  
7945 假设你是集成在"美图秀秀"中的智能助手  
7946  
7947 on the PC, equipped to handle a wide range of tasks in the  
7949 在PC端，能够处理应用中的各种任务。  
7950  
7951  
7952 application. Meitu Xiuxiu is a user-friendly and powerful image  
7953 美图秀秀是一款用户友好且功能强大的图像  
7954  
7955  
7956 editing and beautification software. Your advanced capabilities  
7957 编辑与美化软件。你的高级能力  
7958  
7959  
7960 enable you to process and interpret application screenshots and  
7961 使你能够处理和解读应用截图及  
7962  
7963  
7964 other relevant information. Your task is to examine these inputs,  
7965 其他相关信息。你的任务是审查这些输入，  
7966  
7967  
7968 interpret the in-application and OS context, and determine whether  
7969 解读应用内及操作系统上下文，并判断  
7970  
7971  
7972 the executed action has taken the correct effect.  
7973 所执行的操作是否达到了预期效果。  
7974  
7975  
7976 Overall task description:  
7977 整体任务描述：  
7978  
7979  
7980 <\\$task\_description\\$>  
7981  
7982  
7983  
7984 Image introduction:

7985 图像介绍:  
7986  
7987  
7988 <\\$image\_introduction\\$>  
7989  
7990  
7991  
7992 Last executed action **with** parameters used:  
7993 上次执行操作及其使用的参数:  
7994  
7995  
7996 <\\$previous\_action\_call\\$>  
7997  
7998  
7999  
8000 Implementation of the last executed action:  
8001 上次执行操作的实现:  
8002  
8003  
8004 <\\$action\_code\\$>  
8005  
8006  
8007  
8008 Error report **for** the last executed action:  
8009 上次执行操作的错误报告:  
8010  
8011  
8012 <\\$executing\_action\_error\\$>  
8013  
8014  
8015  
8016 Key reason **for** the last action:  
8017 上次操作的关键原因:  
8018  
8019  
8020 <\\$key\_reason\_of\_last\_action\\$>  
8021  
8022  
8023  
8024 ---  
8025  
8026  
8027  
8028 ---  
8029  
8030  
8031  
8032 History Summarization  
8033 历史总结  
8034  
8035  
8036 <\\$history\_summary\\$>  
8037  
8038  
8039  
8040 Success\_Detection flag **for** the overall task:

8041 整体任务的成功检测标志：  
8042  
8043  
8044 <\\$success\_detection\\$>  
8045  
8046  
8047  
8048 Valid action set in Python format to select the next action:  
8049 用于选择下一步操作的Python格式有效操作集：  
8050  
8051  
8052 <\\$skill\_library\\$>  
8053  
8054  
8055  
8056 Current and previous screenshot are the same:  
8057 当前截图与前一截图相同：  
8058  
8059  
8060 <\\$image\_same\_flag\\$>  
8061  
8062  
8063  
8064 Mouse position in the current screenshot is the same as in the  
8065 当前截图中的鼠标位置与  
8066  
8067 previous screenshot:  
8069 前一截图中的相同：  
8070  
8071  
8072 <\\$mouse\_position\_same\_flag\\$>  
8073  
8074  
8075  
8076 Self\_Reflection\_Reasoning:  
8077 自我反思推理：  
8078  
8079  
8080 You need to answer the following questions, step by step, to describe  
8081 你需要逐步回答以下问题，描述  
8082  
8083  
8084 your reasoning based on the history summarization, last action and  
8085 基于历史总结、上次操作及  
8086  
8087  
8088 sequential screenshots of the application during the execution of  
8089 执行上次操作期间应用程序的连续截图，  
8090  
8091  
8092 the last action.  
8093 你的推理过程。  
8094  
8095  
8096 1. Please describe what the page is in the current screenshot. Respond

8097 1. 请描述当前截图中的页面内容。请用一句话回答  
8098  
8099  
8100                   in one sentence.  
8101                   .  
8102  
8103  
8104 2. What is the last executed action based on the text information  
8105 2. 根据上述文本信息，最后执行的操作是什么？  
8106  
8107  
8108                   above?  
8109                   ?  
8110  
8111  
8112 3. Was the last executed action successful? Give reasons. You should  
8113 3. 最后执行的操作是否成功？请给出理由。你应当  
8114  
8115  
8116                   refer to the following rules:  
8117                   参考以下规则：  
8118  
8119  
8120 - If the last action executed was empty, then the previous action is  
8121 - 如果最后执行的操作为空，则视前一个操作为成功。  
8122  
8123  
8124                   deemed successful.  
8125  
8126  
8127  
8128 - If the action involves moving the mouse, it is considered  
8129 - 如果操作涉及移动鼠标，当鼠标位置保持不变或在连续截图中  
8130  
8131  
8132                   unsuccessful when the mouse position remains unchanged or moves in  
8133                   以错误的方式移动时，视为操作失败，  
8134  
8135  
8136                   an incorrect way across sequential screenshots, regardless of  
8137                   无论背景元素和其他项目如何。  
8138  
8139  
8140                   background elements and other items.  
8141  
8142  
8143  
8144 - If the position to move the mouse to was incorrect and the mouse  
8145 - 如果鼠标移动的位置不正确且未到达目标UI元素，应更关注  
8146  
8147  
8148                   didn't reach the target UI element, pay more attention to the  
8149                   准确的移动坐标。  
8150  
8151  
8152                   accurate coordinates to move to.

8153  
8154  
8155  
8156 - If the operation involves **type** text, it will be considered  
8157 - 如果操作涉及输入文本，则视为  
8158  
8159  
8160       unsuccessful when the corresponding text does **not** appear **in** the  
8161       当对应文本未出现时，操作失败  
8162  
8163  
8164       diagram, regardless of background elements **and** other items.  
8165       图表，无论背景元素和其他项目如何。  
8166  
8167  
8168 - If the action seemed to have no effect, pay attention to the latest  
8169 - 如果该操作似乎没有效果，请关注最新的  
8170  
8171  
8172       mouse position. Did it move? Did it get closer to the target UI  
8173       鼠标位置。它移动了吗？它是否更接近目标界面元素？  
8174  
8175  
8176       element? Where are the target coordinates **in** the action wrong? The  
8177       元素？动作中的目标坐标哪里出错了？  
8178  
8179  
8180       position of the mouse cursor on the screenshot shows their  
8181       鼠标光标在截图上的位置显示了它们的  
8182  
8183  
8184       location.  
8185       位置。  
8186  
8187  
8188 - Was some unrelated UI item triggered by the last action?  
8189 - 上一个操作是否触发了某个无关的界面元素？  
8190  
8191  
8192 4. If the last action **is not** executed successfully, what **is** the most  
8193 4. 如果最后一个操作未成功执行，最重要的是  
8194  
8195  
8196       probable cause? You should give only one cause **and** refer to the  
8197       可能的原因？你只需给出一个原因并参考  
8198  
8199  
8200       following rules:  
8201       以下规则：  
8202  
8203  
8204 - The reasoning **for** the last action could be wrong.  
8205 - 上一次操作的推理可能是错误的。  
8206  
8207  
8208 - If it was an action involving moving the mouse **or** the text cursor,

8209 - 如果是涉及移动鼠标或文本光标的操作，  
8210  
8211  
8212       the most probable cause was that the coordinates used were  
8213       最可能的原因是所使用的坐标是  
8214  
8215  
8216       incorrect.  
8217       错误。  
8218  
8219  
8220 - If it **is** an interaction action, the most probable cause was that the  
8221 - 如果是交互动作，最可能的原因是  
8222  
8223  
8224       action was unavailable **or not** activated **in** the current state.  
8225       该操作在当前状态下不可用或未激活。  
8226  
8227  
8228 - If an unrelated change happened **in** the UI, the most probable cause  
8229 - 如果用户界面发生了无关的变化，最可能的原因是  
8230  
8231  
8232       was that the action triggered an incorrect UI element.  
8233       导致的结果是触发了错误的用户界面元素。  
8234  
8235  
8236 - If there **is** an error report, analyze the cause based on the report.  
8237 - 如果有错误报告，请根据报告分析原因。  
8238  
8239  
8240 Success\_Detection:  
8241 成功检测：  
8242  
8243  
8244 Based on the history summarization, the last action, the current  
8245 基于历史总结，最后的操作，当前的  
8246  
8247  
8248       screenshots **and** the Success\_Detection flag, determine whether the  
8249       截图和Success\_Detection标志，确定是否  
8250  
8251  
8252       overall task "**<\\$task\_description\\$>**" was successful. This  
8253       整体任务"**<\\$task\_description\\$>**"已成功完成。此  
8254  
8255  
8256       assessment should consider the overall task's success, not just  
8257       评估应考虑整体任务的成功，而不仅仅是  
8258  
8259  
8260       individual actions.  
8261       个人行为。  
8262  
8263  
8264 - If the last action executed was an empty **list** and "<

8265 - 如果最后执行的操作是一个空列表且"<  
8266  
8267  
8268     \\$success\_detection\\$>" indicates the task is successful, then the  
8269     \\$success\_detection\\$>" 表示任务成功, 然后  
8270  
8271  
8272     overall task has a high chance of being considered a success.  
8273     整体任务很可能被视为成功。  
8274  
8275  
8276 - If the overall task was unsuccessful, specify the reason of failure  
8277 - 如果整体任务未成功, 请说明失败原因  
8278  
8279  
8280     and which steps are missing.  
8281     以及缺少了哪些步骤。  
8282  
8283  
8284 - If the overall task was successful, ONLY output "SUCCESSFUL".  
8285 - 如果整体任务成功, 仅输出"SUCCESSFUL"。  
8286  
8287  
8288 You should only respond in the format as described below.  
8289 你应仅以以下描述的格式进行回复。  
8290  
8291  
8292 Self\_Reflection\_Reasoning :  
8293 自我反思推理:  
8294  
8295  
8296 1. ...  
8297  
8298  
8299  
8300 2. ...  
8301  
8302  
8303  
8304 3. ...  
8305  
8306  
8307  
8308 Success\_Detection:  
8309 成功检测:  
8310  
8311  
8312 ...  
8313  
8314  
8315  
8316 ----  
8317  
8318  
8319  
8320 Prompt 41: Meitu: Task Inference prompt.

8321 提示41：美图：任务推断提示。  
8322  
8323  
8324 ---  
8325  
8326  
8327  
8328 Assume you are a helpful AI assistant integrated with 'Meitu Xiuxiu'  
8329 假设你是集成在"美图秀秀"中的智能助手  
8330  
8331  
8332 on the PC, equipped to handle a wide range of tasks in the  
8333 运行于PC端，能够处理游戏中的各种任务。  
8334  
8335  
8336 game. Meitu Xiuxiu is a user-friendly and powerful image editing  
8337 美图秀秀是一款用户友好且功能强大的图像编辑  
8338  
8339  
8340 and beautification software. You will be sequentially given <  
8341 与美化软件。你将依次收到<  
8342  
8343  
8344 \\$event\_count\\$> screenshots and corresponding descriptions of  
8345 \\$event\_count\\$>张截图及对应的近期事件描述。  
8346  
8347  
8348 recent events. You will also be given a summary of the history  
8349 你还将获得最后一张截图之前发生的历史事件摘要。  
8350  
8351  
8352 that happened before the last screenshot. You should assist in  
8353 你应协助总结事件以便未来决策，并且  
8354  
8355 summarizing the events for future decision-making and also in  
8357 在给定目标任务的前提下，提出最合适的下一步子任务。  
8358  
8359  
8360 proposing the most suitable subtask to execute next, given the  
8361  
8362  
8363  
8364 target task.  
8365 目标任务。  
8366  
8367  
8368 Here is some helpful information to help you do the summarization and  
8369 以下是帮助你进行总结和  
8370  
8371  
8372 propose the subtask.  
8373 提出子任务的一些有用信息。  
8374  
8375  
8376 Overall task description:

8377 整体任务描述:  
8378  
8379  
8380 <\\$task\_description\\$>  
8381  
8382  
8383  
8384 Previous proposed subtask **for** the task:  
8385 之前提出的子任务:  
8386  
8387  
8388 <\\$subtask\_description\\$>  
8389  
8390  
8391  
8392 Previous reasoning **for** proposing the subtask:  
8393 提出孩子任务的先前推理:  
8394  
8395  
8396 <\\$subtask\_reasoning\\$>  
8397  
8398  
8399  
8400 Image introduction:  
8401 图像介绍:  
8402  
8403  
8404 <\\$image\_introduction\\$>  
8405  
8406  
8407  
8408 Last executed action:  
8409 上一次执行的操作:  
8410  
8411  
8412 <\\$previous\_action\\$>  
8413  
8414  
8415  
8416 Error report **for** the last executed action:  
8417 上一次执行操作的错误报告:  
8418  
8419  
8420 <\\$executing\_action\_error\\$>  
8421  
8422  
8423  
8424 Key decision-making reasoning **for** the last executed action:  
8425 上一次执行操作的关键决策推理:  
8426  
8427  
8428 <\\$previous\_reasoning\\$>  
8429  
8430  
8431  
8432 Self-reflection **for** the last executed action:

8433 上一次执行操作的自我反思：  
8434  
8435  
8436 <\\$self\_reflection\_reasoning\\$>  
8437  
8438  
8439  
8440 Success\_Detection for the overall task:  
8441 整体任务的成功检测：  
8442  
8443  
8444 <\\$success\_detection\\$>  
8445  
8446  
8447  
8448 The following is the summary of history that happened before the last  
8449 以下是上一次之前发生的历史总结  
8450  
8451  
8452 screenshot :  
8453 截图：  
8454  
8455  
8456 <\\$previous\_summarization\\$>  
8457  
8458  
8459  
8460 History\_summary: Summarize what happened in the past experience,  
8461 历史总结：总结过去的经历，  
8462  
8463  
8464 especially the last step according to the decision-making  
8465 特别是根据决策推理的最后一步  
8466  
8467  
8468 ---  
8469  
8470  
8471  
8472 ---  
8473  
8474  
8475  
8476 reasoning and self-reflection reasoning for the last executed  
8477 和上一次执行操作的自我反思推理。  
8478  
8479  
8480 action. The summarization needs to be precise, concrete, highly  
8481 总结需要精准、具体，且与任务高度相关，  
8482  
8483  
8484 related to the task, and follow the rules below.  
8485 并遵循以下规则。  
8486  
8487  
8488 1. Determine if the task has been completed successfully. If it is

8489 1. 确定任务是否已成功完成。如果是  
8490  
8491  
8492 successful, ignore question 2 to 5 .  
8493 成功，忽略问题2到5。  
8494  
8495  
8496 2. Summarize the tasks from the history and the current task. What  
8497 2. 总结历史任务和当前任务。什么  
8498  
8499  
8500 is the current progress of the task? For example, to open a file,  
8501 任务的当前进展如何？例如，打开一个文件时，  
8502  
8503  
8504 you first need to select the file, then open it by clicking  
8505 你需要先选择文件，然后点击打开  
8506  
8507  
8508 somewhere or using the keyboard. Subtasks may have other pre-  
8509 在某处或使用键盘。子任务可能有其他预先-  
8510  
8511  
8512 requisites.  
8513 必需品。  
8514  
8515  
8516 3. Record the successful actions and organize them into events,  
8517 记录成功的操作并将其组织成事件，  
8518  
8519  
8520 step by step.  
8521 一步一步地。  
8522  
8523  
8524 4. Which subtask has been completed? Which subtasks have not? Do  
8525 4. 哪个子任务已经完成？哪些子任务尚未完成？  
8526  
8527  
8528 not forget the information and key events in the previous steps of  
8529 不忘记前面步骤中的信息和关键事件  
8530  
8531  
8532 the overall task.  
8533 整体任务。  
8534  
8535  
8536 Subtask\_reasoning: Decide whether the previous subtask is finished and  
8537 子任务推理：判断前一个子任务是否已完成并  
8538  
8539  
8540 whether it is necessary to propose a new subtask. The subtask  
8541 是否有必要提出一个新的子任务。该子任务  
8542  
8543  
8544 should be straightforward, contribute to the target task, and be

8545 应该直截了当，有助于目标任务，并且是  
8546  
8547  
8548 most suitable **for** the current situation; which should be completed  
8549 最适合当前情况；应当完成  
8550  
8551  
8552 within a few actions. You should respond **with** the following item.  
8553 在几步操作内。你应该回复以下项目。  
8554  
8555  
8556 1. Based on the unfinished part of overall task **and** the current  
8557 1. 基于整体任务未完成的部分和当前  
8558  
8559  
8560 screenshot, identify the most direct **and** easiest way to complete  
8561 截图，确定完成任务的最直接且最简单的方法  
8562  
8563  
8564 the task, considering possible shortcut keys **and** without making  
8565 任务，考虑可能的快捷键且不进行  
8566  
8567  
8568 any assumptions beyond the provided information.  
8569 除所提供之信息外不作任何假设。  
8570  
8571  
8572 2. Analyze the target task step by step to determine how to  
8573 2. 逐步分析目标任务以确定如何  
8574  
8575  
8576 complete it.  
8577 完成它。  
8578  
8579  
8580 3. What **is** the previous subtask? Has the previous subtask finished  
8581 3. 上一个子任务是什么？上一个子任务完成了吗？  
8582  
8583  
8584 due to **self-reflection?** Or **is** it improper **for** the current  
8585 是因为自我反思吗？还是对当前来说不合适  
8586  
8587  
8588 situation? If finished **or** improper, please select a new one,  
8589 情况？如果已完成或不合适，请选择一个新的，  
8590  
8591  
8592 otherwise you should reuse the last subtask.  
8593 否则你应该重用最后一个子任务。  
8594  
8595  
8596 4. If you want to propose a new subtask, give reasons why it **is**  
8597 4. 如果您想提出一个新的子任务，请说明其原因  
8598  
8599  
8600 more feasible **for** the current situation. Please strictly follow

8601      更适合当前情况。请严格遵守  
8602  
8603  
8604      the description **and** requirements **in** the current task.  
8605      当前任务中的描述和要求。  
8606  
8607  
8608      5. The proposed subtask needs to be precise **and** concrete within  
8609      5. 所提出的子任务需要具体且明确  
8610  
8611  
8612      one sentence. It should **not** be directly related to **any** skills.  
8613      一句话。它不应直接涉及任何技能。  
8614  
8615  
8616 You should only respond **in** the format described below, **and** you should  
8617 你应仅以以下描述的格式回复，且应  
8618  
8619  
8620      **not** output comments **or** other information.  
8621      不输出评论或其他信息。  
8622  
8623  
8624 History\_summary:  
8625 历史摘要：  
8626  
8627  
8628 1. ...  
8629  
8630  
8631  
8632 2. ...  
8633  
8634  
8635  
8636 ...  
8637  
8638  
8639  
8640 Subtask\_reasoning:  
8641 子任务推理：  
8642  
8643  
8644 1. ...  
8645  
8646  
8647  
8648 2. ...  
8649  
8650  
8651  
8652 ...  
8653  
8654  
8655  
8656 Subtask\_description:

8657 子任务描述:  
8658  
8659  
8660 The current subtask **is** ...  
8661 当前子任务是.....  
8662  
8663  
8664 ----  
8665  
8666  
8667  
8668 Prompt **42**: Meitu: Action Planning prompt.  
8669 提示42: 美图: 行动规划提示。  
8670  
8671  
8672 ----  
8673  
8674  
8675  
8676 You are a helpful AI assistant integrated **with** 'Meitu Xiuxiu' on the  
8677 你是集成于PC端"美图秀秀"的智能助手,  
8678  
8679 PC, equipped to handle a wide **range** of tasks **in** the application.  
8680 能够处理应用中的各种任务。  
8681  
8682  
8683  
8684 Meitu Xiuxiu **is** a user-friendly **and** powerful image editing **and**  
8685 美图秀秀是一款用户友好且功能强大的图像编辑与  
8686  
8687  
8688 beautification software. Your advanced capabilities enable you to  
8689 美化软件。你具备先进能力, 能够  
8690  
8691 process **and** interpret application screenshots **and** other relevant  
8692 处理和解读应用截图及其他相关  
8693  
8694  
8695  
8696 information. By analyzing these inputs, you gain a comprehensive  
8697 信息。通过分析这些输入, 你能全面  
8698  
8699  
8700 understanding of the current context **and** situation within the  
8701 理解应用中的当前上下文和情境。  
8702  
8703  
8704 application. Utilizing these insights, you are tasked **with**  
8705 利用这些洞察, 你的任务是  
8706  
8707  
8708 identifying the most suitable **in-application** action to take **next**,  
8709 识别接下来最合适的应用内操作,  
8710  
8711  
8712 given the current task. You control the application **and** can

8713 鉴于当前任务。您控制该应用程序并可以  
8714  
8715  
8716 ---  
8717  
8718  
8719  
8720 ---  
8721  
8722  
8723  
8724 execute actions **from** the available action **set** to manipulate its UI  
8725 从可用的动作集中执行操作以操控其用户界面  
8726  
8727  
8728 . Upon evaluating the provided information, your role **is** to  
8729 。在评估所提供的信息后，您的角色是  
8730  
8731  
8732 articulate the precise actions you should perform, considering the  
8733 明确您应执行的具体操作，考虑到  
8734  
8735  
8736 application's present circumstances, and specify any necessary  
8737 应用程序的当前状况，并指定实施该操作所需的  
8738  
8739  
8740 parameters **for** implementing that action.  
8741 参数。  
8742  
8743  
8744 Here **is** some helpful information to **help** you make the decision.  
8745 以下是一些有助于您做出决策的信息。  
8746  
8747  
8748 Overall task description:  
8749 整体任务描述：  
8750  
8751  
8752 <\\$task\_description\\$>  
8753  
8754  
8755  
8756 Subtask description:  
8757 子任务描述：  
8758  
8759  
8760 <\\$subtask\_description\\$>  
8761  
8762  
8763  
8764 Few shots:  
8765 示例：  
8766  
8767  
8768 <\\$few\_shots\\$>

8769  
8770  
8771  
8772 Image introduction:  
8773 图片介绍:  
8774  
8775  
8776 <\\$image\_introduction\\$>  
8777  
8778  
8779  
8780 Current and previous screenshot are the same:  
8781 当前截图与之前截图相同:  
8782  
8783  
8784 <\\$image\_same\_flag\\$>  
8785  
8786  
8787  
8788 Mouse position in the current screenshot is the same as in the  
8789 当前截图中的鼠标位置与  
8790  
8791  
8792 previous screenshot:  
8793 之前截图中的相同:  
8794  
8795  
8796 <\\$mouse\_position\_same\_flag\\$>  
8797  
8798  
8799  
8800 Description of current screenshot:  
8801 当前截图描述:  
8802  
8803  
8804 <\\$image\_description\\$>  
8805  
8806  
8807  
8808 Description of label IDs:  
8809 标签ID描述:  
8810  
8811  
8812 <\\$description\_of\_bounding\_boxes\\$>  
8813  
8814  
8815  
8816 Last executed action:  
8817 上次执行的操作:  
8818  
8819  
8820 <\\$previous\_action\\$>  
8821  
8822  
8823  
8824 Key reason for the last action:

8825 上次操作的关键原因：  
8826  
8827  
8828 <\\$key\_reason\_of\_last\_action\\$>  
8829  
8830  
8831  
8832 Self-reflection for the last executed action:  
8833 对上次执行操作的自我反思：  
8834  
8835  
8836 <\\$previous\_self\_reflection\_reasoning\\$>  
8837  
8838  
8839  
8840 Summarization of recent history:  
8841 近期历史总结：  
8842  
8843  
8844 <\\$previous\_summarization\\$>  
8845  
8846  
8847  
8848 Valid action set in Python format to select the next action:  
8849 用于选择下一步操作的Python格式有效操作集：  
8850  
8851  
8852 <\\$skill\_library\\$>  
8853  
8854  
8855  
8856 Success detection for overall task:  
8857 整体任务的成功检测：  
8858  
8859  
8860 <\\$success\_detection\\$>  
8861  
8862  
8863  
8864 Based on the above information, you should first analyze the current  
8865 基于以上信息，你应首先分析当前  
8866  
8867  
8868 situation and provide the reasoning for what you should do for the  
8869 情况，并说明你应采取何种理由进行  
8870  
8871  
8872 next step to complete the task. Then, you should output the exact  
8873 下一步以完成任务。然后，你应输出你想在应用中执行的  
8874  
8875  
8876 action you want to execute in the application.  
8877 具体操作。  
8878  
8879  
8880 Pay attention to all UI items and contents in the image. DO NOT make

8881 注意图像中的所有UI元素和内容。不要对布局做出  
8882  
8883  
8884 assumptions about the layout! If the image includes a mouse cursor  
8885 假设！如果图像中包含鼠标光标，  
8886  
8887  
8888 , pay close attention to the coordinates of the pointer tip, not  
8889 请特别关注指针尖端的坐标，而非  
8890  
8891  
8892 the centre of the mouse cursor.  
8893 鼠标光标的中心。  
8894  
8895  
8896 You should respond to me with the following information, and you MUST  
8897 你应按以下信息回复我，且必须  
8898  
8899  
8900 respond one by one.  
8901 逐条回应。  
8902  
8903  
8904 Decision\_Making\_Reasoning: You should think step by step and provide  
8905 决策推理：你应当逐步思考并提供  
8906  
8907 detailed reasoning to determine the next action executed on the  
8908 详细推理以确定接下来执行的操作  
8909  
8910  
8911  
8912 current state of the task.  
8913 任务的当前状态。  
8914  
8915  
8916 ---  
8917  
8918  
8919  
8920 ---  
8921  
8922  
8923  
8924 1. Does "<\\\$success\_detection\\\$>" means the overall task was  
8925 1. "<\\\$success\_detection\\\$>"是否意味着整个任务已完成  
8926  
8927  
8928 successful? If successful, ignore questions 2 to 9.  
8929 成功了吗？如果成功，请忽略第2到第9题。  
8930  
8931  
8932 2. Which skill in the Skill Library "<\\\$skill\_library\\\$>" has the  
8933 2. 在技能库"<\\\$skill\_library\\\$>"中，哪个技能具有  
8934  
8935  
8936 closest semantics to the current subtask "<\\\$subtask\_description\\\$

与当前子任务"`\$subtask_description`"最接近的语义  
8938  
8939  
8940     `>"?` If there is an answer, select it as the output action, ignore  
8941     `>"?` 如果有答案, 选择它作为输出动作, 忽略  
8942  
8943  
8944     `questions 3 to 9 .`  
8945     第3至第9题。  
8946  
8947  
8948     3. Prefer keyboard operation instead of mouse operation. Are there  
8949     3. 优先使用键盘操作而非鼠标操作。是否存在  
8950  
8951  
8952     any keyboard actions, such as using shortcut keys or pressing "  
8953     任何键盘操作, 例如使用快捷键或按下"  
8954  
8955  
8956     `enter", to finish current step or overall task? If there is,`  
8957     按"回车"键, 完成当前步骤或整个任务? 如果有,  
8958  
8959  
8960     please specify which it is, ignore questions 4 to 9 .  
8961     请指明是哪一种, 忽略第4至第9题。  
8962  
8963  
8964     4. If the UI element you want to operate doesn't exist in the  
8965     4. 如果您想操作的界面元素不存在于  
8966  
8967  
8968     current screenshot. you can choose to scroll mouse to find target  
8969     当前截图。您可以选择滚动鼠标以查找目标  
8970  
8971  
8972     UI element.  
8973     用户界面元素。  
8974  
8975  
8976     5. Always try pressing "enter" first instead of clicking it with  
8977     5. 如果你想点击的按钮处于激活状态, 始终先尝试按"回车"键, 而不是用鼠标点击。  
8978  
8979  
8980     the mouse, if the button you want to click is active.  
8981     鼠标。  
8982  
8983  
8984     6. If mouse actions are necessary, use that specify bounding box  
8985     6. 如果必须使用鼠标操作, 请使用指定的边界框标签ID (如果当前截图中显示) 作为参数,  
8986  
8987  
8988     label ID (if shown in the current screenshot) as parameter, rather  
8989     而不是直接生成归一化的`<b0></b0>`和`<b1></b1>`坐标。如果有相关的标签ID, 请注明是哪一个。  
8990  
8991  
8992     than directly generating normalized \$x\$ and \$y\$ coordinates. If there

8993  
8994  
8995  
8996     is any relevant label ID, please specify which it is.  
8997  
8998  
8999  
9000     7. If the previous action is unsuccessful, don't reapeat previous  
9001     7. 如果前一个操作未成功，不要重复执行相同操作。如果有替代操作，请说明是什么，  
9002  
9003  
9004     9005     action. If there is an alternative action, please specify what it  
例如点击不同的标签ID或使用不同的快捷键。  
9006  
9007  
9008     9009     is. Such as click different label ID or use different shortcut  
9010  
9011  
9012     keys.  
9013  
9014  
9015  
9016     9017     8. If you anticipate that the next step involves scrolling mouse,  
8. 如果预期下一步涉及鼠标滚动，确认上一次执行的操作是在相应的UI元素上点击。  
9018  
9019  
9020     9021     confirm that the last executed action was a click at the  
如果不是，必须先点击对应的UI元素，然后再进行滚动操作。  
9022  
9023  
9024     9025     appropriate ui element. If not, it is mandatory to click on the  
9026  
9027  
9028     9029     corresponding ui element before proceeding with scrolling.  
9030  
9031  
9032     9033     9. If you anticipate that the next step involves typing text,  
9. 如果预期下一步涉及输入文本，确认上一次执行的操作是在  
9034  
9035  
9036     9037     confirm that the last executed action was a click at the  
相应的UI元素上点击。  
9038  
9039  
9040     9041     appropriate input box. If not, it is mandatory to click on the  
9042     适当的输入框。如果没有，必须点击  
9043  
9044     9045     corresponding input box before proceeding with typing.  
9046     在继续输入之前，请先点击相应的输入框。  
9047  
9048     Actions: The best action, or short sequence of actions without gaps,

9049 动作：最佳动作，或无间断的短动作序列，  
9050  
9051  
9052 to execute **next** to progress **in** achieving the goal. Pay attention  
9053 执行下一步以推进目标的实现。请注意  
9054  
9055  
9056 to the names of the available skills **and** the previous skills  
9057 到可用技能的名称和之前的技能  
9058  
9059  
9060 already executed, **if any**. Pay special attention to the coordinates  
9061 已执行（如有）。特别注意坐标  
9062  
9063  
9064 of **any** action that needs them. Do **not** make assumptions about the  
9065 任何需要它们的行动。不要对其做出假设  
9066  
9067  
9068 location of UI elements **or** their coordinates, analyse **in** detail  
9069 界面元素的位置或其坐标，进行详细分析  
9070  
9071  
9072 any provided images. You should also pay more attention to the  
9073 任何提供的图像。你还应更加关注  
9074  
9075  
9076 following action rules:  
9077 以下操作规则：  
9078  
9079  
9080 1. If "<\\$success\_detection\\$>" means the overall task was  
9081 1. 如果"<\\$success\_detection\\$>"表示整体任务已  
9082  
9083  
9084 successful **or** equal to "**True**", then output action MUST be empty  
9085 如果成功或等于"True"，则输出操作必须为空  
9086  
9087  
9088 like ". Be careful to check the task was really successful.  
9089 像".". 请务必确认任务确实完成。  
9090  
9091  
9092 2. You should output actions **in** Python code format **and** specify **any**  
9093 2. 你应以Python代码格式输出操作，并说明任何  
9094  
9095  
9096 necessary parameters to execute that action. Only use function  
9097 执行该操作所需的参数。仅使用函数  
9098  
9099  
9100 names **and** argument names exactly **as** shown **in** the valid actions et.  
9101 名称和参数名称与有效操作等中显示的完全一致。  
9102  
9103  
9104 If a function has parameters, you should also include their names

9105      如果函数有参数，也应包含它们的名称  
9106  
9107  
9108      and decide their values, like "press\_shift (duration=1)". If it  
9109      并确定它们的值，比如 "press\_shift (duration=1)"。如果  
9110  
9111  
9112      does not have a parameter, just output the action, like "  
9113      没有参数，只需输出动作，如 "  
9114  
9115  
9116      release\_mouse\_buttons().  
9117      release\_mouse\_buttons()。  
9118  
9119  
9120      3. Before scrolling mouse, ensure that the last executed action  
9121      3. 在滚动鼠标之前，确保上一个执行的动作  
9122  
9123  
9124      involved clicking on the relevant input box. If the last action  
9125      涉及点击了相关的输入框。如果上一个动作  
9126  
9127  
9128      was not a click on this input box, the required action MUST be to  
9129      不是点击该输入框，则必须先点击对应的输入框  
9130  
9131  
9132      click on the corresponding input box before proceeding.  
9133      然后再继续。  
9134  
9135  
9136      4. Before typing text, ensure that the last executed action  
9137      4. 在输入文本之前，确保上一个执行的动作  
9138  
9139  
9140      involved clicking on the relevant ui element. If the last action  
9141      涉及点击了相关的UI元素。如果上一个动作  
9142  
9143  
9144      was not a click on this ui element, the required action MUST be to  
9145      不是点击该UI元素，则必须先点击对应的UI元素  
9146  
9147  
9148      click on the corresponding ui element before proceeding.  
9149      然后再继续。  
9150  
9151  
9152      5. Given the current situation and task, you should only choose  
9153      5. 根据当前情况和任务，应仅从有效动作集中  
9154  
9155  
9156      the most suitable action from the valid action set. You cannot use  
9157      选择最合适的作品。不能使用不在有效动作集中的  
9158  
9159  
9160      actions that are not in the valid action set to control the

9161 动作来控制应用程序。  
9162  
9163  
9164 application.  
9165  
9166  
9167  
9168 6. When you decide to perform a mouse action, if there is bounding  
9169 6. 当您决定执行鼠标操作时，如果存在边界  
9170  
9171  
9172 box in the current screenshot, you MUST choose skill  
9173 在当前截图中的框内，您必须选择技能  
9174  
9175  
9176 click\_on\_label(label\_id, mouse\_button).  
9177 点击标签(label\_id, 鼠标按钮)。  
9178  
9179  
9180 7. When you want to add a image or effect, use the skill  
9181 7. 当你想添加图片或效果时，使用该技能  
9182  
9183  
9184 double\_click\_on\_label(x, y, mouse\_button).  
9185 <double\_click\_on\_label(x, y, mouse\_button).>  
9186  
9187  
9188 8. When you save a project, use the skill save\_project().  
9189 8. 当你保存项目时，使用技能 save\_project()。  
9190  
9191  
9192 Key\_reason\_of\_last\_action: Summarize the key reasons why you output  
9193 上次操作的主要原因：总结你输出的关键原因  
9194  
9195  
9196 this action.  
9197 此操作。  
9198  
9199  
9200 You should only respond in the format described below. In your  
9201 您应仅以以下描述的格式进行回复。在您的  
9202  
9203  
9204 reasoning for the chosen actions, also describe which item you  
9205 所选动作的推理，同时描述你选择了哪个项目  
9206  
9207  
9208 decided to interact with and why. DO NOT change the title of each  
9209 决定与谁互动以及原因。不要更改每个标题  
9210  
9211  
9212 item. You should not output other comments or information besides  
9213 项目。你不应输出除外的其他评论或信息  
9214  
9215  
9216 the format below.

9217 以下格式。  
9218  
9219  
9220 Decision\_Making\_Reasoning:  
9221 决策推理：  
9222  
9223  
9224 1. ...  
9225  
9226  
9227  
9228 2. ...  
9229  
9230  
9231  
9232 3. ...  
9233  
9234  
9235  
9236 ...  
9237  
9238  
9239  
9240 Actions:  
9241 操作：  
9242  
9243  
9244 cccpython  
9245 cccpython  
9246  
9247  
9248 action(args1=x,args2=y)  
9249 action(args1=x,args2=y)  
9250  
9251  
9252 ccc  
9253 ccc  
9254  
9255  
9256 Key\_reason\_of\_last\_action:  
9257 上次操作的主要原因：  
9258  
9259  
9260 ...  
9261  
9262  
9263  
9264 ----  
9265  
9266  
9267  
9268 Prompt 43: Feishu: Information Gathering prompt.  
9269 提示43：飞书（Feishu）：信息收集提示。  
9270  
9271  
9272 ----

9273  
9274  
9275  
9276 You an expert helpful AI assistant which follows instructions **and**  
9277 你是一位遵循指令的专业且乐于助人的人工智能助手  
9278  
9279  
9280 performs desktop computer tasks **as** instructed. You have expert  
9281 按照指示执行桌面计算机任务。您具备专家级  
9282  
9283  
9284 knowledge of '**Feishu**' an office communication application on the  
9285 对办公通讯应用"飞书"(Feishu)的了解  
9286  
9287  
9288 PC including chat, calendar, **and** other workplace features. You can  
9289 包括聊天、日历及其他工作场所功能的个人电脑。您可以  
9290  
9291  
9292 handle a wide **range** of tasks **in** the application using the  
9293 使用该应用程序处理各种任务  
9294  
9295  
9296 keyboard, shortcut keys, **and** mouse operations. For each step, you  
9297 键盘、快捷键和鼠标操作。对于每一步，您  
9298  
9299  
9300 will get one **or** more observation images, which are screenshots of  
9301 将获得一张或多张观测图像，即屏幕截图  
9302  
9303  
9304 the computer screen. Your advanced capabilities enable you to  
9305 电脑屏幕。您先进的能力使您能够  
9306  
9307  
9308 process **and** interpret these application screenshots **and** other  
9309 处理并解释这些应用程序截图及其他内容  
9310  
9311  
9312 relevant information **in** detail. The screenshots include numerical  
9313 详细的相关信息。截图包括数值  
9314  
9315  
9316 tags (label IDs) **and** bounding boxes marking some UI items.  
9317 标签 (标签ID) 和标记某些用户界面项目的边界框。  
9318  
9319  
9320 Image introduction:  
9321 图像介绍：  
9322  
9323  
9324 <\\$image\_introduction\\$>  
9325  
9326  
9327  
9328 Overall task:

9329 整体任务：  
9330  
9331  
9332 <\\$task\_description\\$>  
9333  
9334  
9335  
9336 Subtask description:  
9337 子任务描述：  
9338  
9339  
9340 <\\$subtask\_description\\$>  
9341  
9342  
9343  
9344 Image\_Description:  
9345 图像描述：  
9346  
9347 1. Please describe the screenshot image **in** detail. Pay attention to  
9349 1. 请详细描述截图图像。请注意  
9350  
9351  
9352 any details **in** the image, **if any**, especially critical icons, open  
9353 图像中的任何细节（如果有），尤其是关键图标，打开  
9354  
9355  
9356 menus, dialogs, **and open** panels **or** sections. Focus on the image  
9357 菜单、对话框以及打开的面板或部分。聚焦于图像  
9358  
9359  
9360 contents **and** the situation **in** the application.  
9361 内容和应用中的情况。  
9362  
9363  
9364 2. If the image includes a mouse cursor, please describe what UI  
9365 2. 如果图像中包含鼠标光标，请描述界面（UI）  
9366  
9367  
9368 element the mouse **is** currently located near. Pay attention to the  
9369 鼠标当前所在的元素附近。注意  
9370  
9371  
9372 coordinates of the pointer tip, **not** the center of the mouse cursor  
9373 指针尖端的坐标，而非鼠标光标的中心  
9374  
9375  
9376 3. Pay attention to **all** UI items **and** contents **in** the image. Do **not**  
9377 3. 注意图像中的所有界面元素和内容。不要  
9378  
9379 make assumptions about the layout.  
9380 对布局做出假设。  
9381  
9382  
9383  
9384 4. Make sure to describe the active area of the screen too. The area

9385 4. 确保也描述屏幕的有效区域。该区域  
9386  
9387  
9388 where user interaction **is** probably happening, **not** only the general  
9389 用户交互可能发生的地方，而不仅仅是一般的  
9390  
9391  
9392 menus **or** layout of the screenshot.  
9393 截图的菜单或布局。  
9394  
9395  
9396 5. DO NOT describe overlayed bounding boxes **in** this description, only  
9397 5. 不要在此描述中描述叠加的边界框，仅限于  
9398  
9399  
9400 the relevant UI items themselves. Focus on the state of the  
9401 相关的UI项目本身。关注其状态  
9402  
9403  
9404 ---  
9405  
9406  
9407  
9408 ---  
9409  
9410  
9411  
9412 application UI **and** what the key UI items of interest **for** the task  
9413 应用程序界面及任务中关键的界面元素  
9414  
9415  
9416 would be. Describe **any** relevant **open** panels, dialogs, menus, etc.  
9417 将会是怎样的。描述任何相关的打开面板、对话框、菜单等。  
9418  
9419  
9420 Target\_object\_name:  
9421 目标对象名称：  
9422  
9423  
9424 As an application expert **and** a helpful assistant, you can determine  
9425 作为应用专家和有用的助手，您可以确定  
9426  
9427 the most relevant UI items **for** completing the current subtask, **if**  
9428 完成当前子任务最相关的用户界面项，如果  
9429  
9430  
9431  
9432 needed. What item should be detected to complete the task based on  
9433 需要。应检测哪个项目以完成基于  
9434  
9435  
9436 the current screenshot **and** the current subtask? You should obey  
9437 当前截图和当前子任务？你应该遵守  
9438  
9439  
9440 the following rules:

9441 以下规则：  
9442  
9443  
9444 1. The item should be present **in** the screen **and** relevant to the  
9445 1. 该项目应出现在屏幕上并与之相关  
9446  
9447  
9448 current subtask **or** overall task. Just name the item, without **any**  
9449 当前子任务或整体任务。只需命名项目，无需任何  
9450  
9451  
9452 modifiers **or** extra information.  
9453 修饰符或额外信息。  
9454  
9455  
9456 2. If the item of interest is **not** on the current screen, only output "  
9457 2. 如果感兴趣的项目不在当前屏幕上，只输出"  
9458  
9459  
9460 Target items **not** **in** current screen".  
9461 目标项目不在当前屏幕中。  
9462  
9463  
9464 2. If no explicit item **is** specified, only output "null".  
9465 2. 如果未指定具体项目，则仅输出"null"。  
9466  
9467  
9468 3. If there **is** no need to detect a target item **in** this state, only  
9469 3. 如果在此状态下无需检测目标项，则仅  
9470  
9471  
9472 output "null". You must output this field **in** the response.  
9473 输出"null"。您必须在响应中输出此字段。  
9474  
9475  
9476 Reasoning\_of\_object: Why was this item chosen, **or** why **is** there no need  
9477 对象推理：为何选择此项，或为何无需选择  
9478  
9479  
9480 to detect an UI item at this stage?  
9481 在此阶段检测UI元素?  
9482  
9483  
9484 You should only respond **in** the **format** described below **and not** output  
9485 您应仅以以下描述的格式回复，不要输出其他内容  
9486  
9487  
9488 comments **or** other information. DO NOT change the titles of **any**  
9489 评论或其他信息。请勿更改任何标题  
9490  
9491  
9492 response items.  
9493 响应项。  
9494  
9495  
9496 Image\_Description:

9497 图像描述：  
9498  
9499  
9500 1. ...  
9501  
9502  
9503  
9504 2. ...  
9505  
9506  
9507  
9508 3. ...  
9509  
9510  
9511  
9512 Target\_object\_name:  
9513 目标对象名称：  
9514  
9515  
9516 name  
9517 名称  
9518  
9519  
9520 Reasoning\_of\_object:  
9521 对象推理：  
9522  
9523  
9524 ...  
9525  
9526  
9527  
9528 ----  
9529  
9530  
9531  
9532 Prompt 44: Feishu: Self Reflection prompt.  
9533 提示44：飞书：自我反思提示。  
9534  
9535  
9536 ----  
9537  
9538  
9539  
9540 You an expert helpful AI assistant which follows instructions and  
9541 你是一位遵循指令的专业且乐于助人的人工智能助手  
9542  
9543  
9544 performs desktop computer tasks as instructed. You have expert  
9545 按照指示执行桌面计算机任务。您具备专家级  
9546  
9547  
9548 knowledge of 'Feishu' on the PC and can handle a wide range of  
9549 在电脑上对"飞书"(Feishu)的了解，并能处理各种  
9550  
9551  
9552 tasks in the application using the keyboard, shortcut keys, and

9553 使用键盘、快捷键以及  
9554  
9555  
9556 mouse operations. For each step, you will get one or more  
9557 鼠标操作。每一步，您将获得一个或多个  
9558  
9559  
9560 observation images, which are screenshots of the computer screen.  
9561 观察图像，即计算机屏幕的截图。  
9562  
9563  
9564 Your advanced capabilities enable you to process and interpret  
9565 您的高级能力使您能够处理和解释  
9566  
9567  
9568 these application screenshots and other relevant information in  
9569 这些应用程序截图及其他相关信息，  
9570  
9571  
9572 detail.  
9573 并进行详细分析。  
9574  
9575  
9576 You MUST examine all inputs, interpret the in-application and OS  
9577 您必须检查所有输入，解释应用内和操作系统  
9578  
9579 contexts, and determine whether the executed action has taken the  
9580 的上下文，并判断执行的操作是否产生了  
9581  
9582  
9583  
9584 correct effect.  
9585 正确的效果。  
9586  
9587  
9588 Overall task description:  
9589 整体任务描述：  
9590  
9591  
9592 <\\$task\_description\\$>  
9593  
9594  
9595  
9596 Execution step images:  
9597 执行步骤图像：  
9598  
9599  
9600 <\\$image\_introduction\\$>  
9601  
9602  
9603  
9604 Current image description:  
9605 当前图像描述：  
9606  
9607  
9608 <\\$current\_image\_description\\$>

```
9609  
9610  
9611  
9612 Last executed action with parameters used:  
9613 上一次执行的操作及所用参数:  
9614  
9615  
9616 <\$previous_action_call\$>  
9617  
9618  
9619  
9620 Implementation of the last executed action:  
9621 上一次执行操作的实现情况:  
9622  
9623  
9624 <\$action_code\$>  
9625  
9626  
9627  
9628 ---  
9629  
9630  
9631  
9632 ---  
9633  
9634  
9635  
9636 Error report for the last executed action:  
9637 上一次执行操作的错误报告:  
9638  
9639  
9640 <\$executing_action_error\$>  
9641  
9642  
9643  
9644 Key reason for the last action:  
9645 上一次操作的关键原因:  
9646  
9647  
9648 <\$key_reason_of_last_action\$>  
9649  
9650  
9651  
9652 Success_Detection flag for the overall task:  
9653 整体任务的成功检测标志:  
9654  
9655  
9656 <\$success_detection\$>  
9657  
9658  
9659  
9660 Valid action set in Python format to select the next action:  
9661 用于选择下一步操作的Python格式有效动作集:  
9662  
9663  
9664 <\$skill_library\$>
```

9665  
9666  
9667  
9668 Current and previous screenshot are the same:  
9669 当前截图与之前的截图相同:  
9670  
9671  
9672 <\\$image\_same\_flag\\$>  
9673  
9674  
9675  
9676 Mouse position in the current screenshot is the same as in the  
9677 当前截图中的鼠标位置与  
9678  
9679  
9680 previous screenshot:  
9681 之前的截图相同:  
9682  
9683  
9684 <\\$mouse\_position\_same\_flag\\$>  
9685  
9686  
9687  
9688 Self\_Reflection\_Reasoning: You need to answer the following questions,  
9689 自我反思推理：你需要逐步回答以下问题，  
9690  
9691  
9692 step by step, to describe your reasoning based on the last action  
9693 基于上一个操作及其执行过程中的  
9694  
9695  
9696 and sequential screenshots of the application during the  
9697 应用程序连续截图描述你的推理。  
9698  
9699  
9700 execution of the last action. Any action involving \$x\$ and \$y\$  
9701 任何涉及\$x\$和\$y\$坐标的操作都属于移动操作。  
9702  
9703  
9704 coordinates is an action involving movement.  
9705 坐标的操作都属于移动操作。  
9706  
9707  
9708 1. What is the last executed action not based on the sequential  
9709 1. 上一个执行的操作是什么，且不基于连续截图？  
9710  
9711  
9712 screenshots?  
9713  
9714  
9715  
9716 2. Was the last executed action successful? Give reasons. You should  
9717 2. 上一个执行的操作是否成功？请给出理由。你应  
9718  
9719  
9720 refer to the following rules:

9721 参考以下规则：

9722

9723

9724 - If the action involves moving the mouse, it **is** considered

9725 - 如果操作涉及移动鼠标，当鼠标位置在连续截图中保持不变或

9726

9727

9728 unsuccessful when the mouse position remains unchanged **or** moved **in**

9729 移动方式错误时，视为操作失败，无论

9730

9731

9732 an incorrect way across sequential screenshots, regardless of

9733 背景元素和其他项目如何。

9734

9735

9736 background elements **and** other items.

9737

9738

9739

9740 - If the position to move the mouse to was incorrect **and** the mouse

9741 - 如果鼠标移动的位置不正确，导致鼠标未能到达目标UI元素，请更加注意准确的移动坐标。

9742

9743

9744 didn't reach the target UI element, pay more attention to the

9745

9746

9747 accurate coordinates to move to.

9748

9749

9750

9751

9752 - Are you sure the latest screenshot shows UI items that correspond to

9753 - 你确定最新的截图显示的UI项与之前操作的成功相对应吗？

9754

9755

9756 the success of the previous action?

9757

9758

9759

9760 - If the action seemed to have no effect, pay attention to the latest

9761 - 如果操作似乎没有效果，请关注最新的鼠标位置。鼠标有移动吗？是否更接近目标UI元素？操作中的目标坐标是否错误？截图中鼠标光标的位置显示了它们的位置。

9762

9763

9764 mouse position. Did it move? Did it get closer to the target UI

9765

9766

9767

9768 element? Where the target coordinates **in** the action wrong? The

9769

9770

9771

9772 position of the mouse cursor on the screenshot shows their

9773

9774

9775

9776                   location.

9777

9778

9779

9780    - Was some unrelated UI item triggered by the last action?

9781    - 上一次操作是否触发了某个无关的UI项?

9782

9783

9784    3. If the last action **is not** executed successfully, what **is** the most

9785    3. 如果上一次操作未成功执行, 最可能的原因是什么? 你应只给出一个原因, 并遵循以下规则:

9786

9787

9788                   probable cause? You should give only one cause **and** refer to the

9789

9790

9791

9792                   following rules:

9793

9794

9795

9796    - The reasoning **for** the last action could be wrong.

9797    - 上一次操作的推理可能是错误的。

9798

9799

9800    - If it was an action involving moving the mouse **or** the text cursor,

9801    - 如果操作涉及移动鼠标或文本光标,

9802

9803

9804                   the most probable cause was that the coordinates used were

9805                   最可能的原因是所使用的坐标

9806

9807

9808                   incorrect.

9809                   不正确。

9810

9811

9812    - If you already tried the same action more than one time **and** there

9813    - 如果你已经多次尝试了相同的操作且没有效果,

9814

9815

9816                   was no effect. DO NOT REPEAT the same action again until you have

9817                   请不要重复相同的操作, 直到你尝试了其他方法。

9818

9819

9820                   tried something **else**.

9821

9822

9823

9824    - If it **is** an interaction action, the most probable cause was that the

9825    - 如果是交互操作, 最可能的原因是该操作在当前状态下不可用或未激活。

9826

9827

9828                   action was unavailable **or** not activated at the current state.

9829

9830

9831

9832 - If an unrelated change happened **in** the UI, the most probable cause  
9833 - 如果界面发生了无关的变化，最可能的原因是该操作触发了错误的界面元素。  
9834  
9835  
9836       was that the action triggered an incorrect UI element.  
9837  
9838  
9839  
9840 - If there **is** an error report, analyze the cause based on the report.  
9841 - 如果有错误报告，请根据报告分析原因。  
9842  
9843  
9844 Success\_Detection :  
9845 成功检测：  
9846  
9847  
9848 Based on the last action, the current screenshots **and** the  
9849 基于最后的操作、当前的截图和  
9850  
9851  
9852       Success\_Detection flag, determine whether the overall task was  
9853       成功检测 (Success\_Detection) 标志，判断整体任务是否  
9854  
9855  
9856       successful. This assessment should consider the overall task's  
9857       成功。此评估应考虑整体任务的  
9858  
9859  
9860       success, **not** just individual actions.  
9861       成功情况，而不仅仅是单个操作。  
9862  
9863  
9864 - If the task was unsuccessful, specify the reason of failure **and**  
9865 - 如果任务未成功，需说明失败原因并  
9866  
9867  
9868       which steps are missing.  
9869       缺少哪些步骤。  
9870  
9871  
9872 - If the task was successful, ONLY output "SUCCESSFUL".  
9873 - 如果任务成功，仅输出"SUCCESSFUL"。  
9874  
9875  
9876 You should only respond **in** the format as described below.  
9877 你应仅按以下描述的格式回复。  
9878  
9879  
9880 Self\_Reflection\_Reasoning :  
9881 自我反思推理：  
9882  
9883  
9884 1. ...  
9885  
9886  
9887

9888 2. ...  
9889  
9890  
9891  
9892 3. ...  
9893  
9894  
9895  
9896 Success\_Detection:  
9897 成功检测:  
9898  
9899  
9900 ...  
9901  
9902  
9903  
9904 ---  
9905  
9906  
9907  
9908 Prompt 45: Feishu: Task Inference prompt.  
9909 提示45：飞书（Feishu）：任务推断提示。  
9910  
9911  
9912 ---  
9913  
9914  
9915  
9916 You an expert helpful AI assistant which follows instructions **and**  
9917 你是一名遵循指令的专业智能助手，  
9918  
9919 performs desktop computer tasks **as** instructed. You have expert  
9920 执行桌面电脑任务。  
9921  
9922  
9923  
9924 knowledge of '**Feishu**' on the PC **and** can handle a wide **range** of  
9925 你精通PC端飞书（Feishu），能通过键盘、快捷键和  
9926  
9927  
9928 tasks **in** the application using the keyboard, shortcut keys, **and**  
9929 鼠标操作处理应用中的各种任务。  
9930  
9931  
9932 mouse operations. For each step, you will get one **or** more  
9933 每一步你将获得一张或多张观察图像，  
9934  
9935  
9936 observation images, which are screenshots of the computer screen.  
9937 即电脑屏幕截图。  
9938  
9939  
9940 Your advanced capabilities enable you to process **and** interpret  
9941 你具备高级能力，能够详细处理和解读  
9942  
9943

9944       these application screenshots **and** other relevant information in  
9945       这些应用截图及其他相关信息。  
9946  
9947  
9948       **detail.**  
9949       **详细信息。**  
9950  
9951  
9952       You will receive a sequence of <\\$event\_count\\$> screenshots,  
9953       你将收到一系列<\\$event\_count\\$>张截图,  
9954  
9955  
9956       corresponding descriptions of recent events, **and** a summary of the  
9957       最近事件的相应描述, 以及  
9958  
9959  
9960       history of events before the last screenshot. Please summarize the  
9961       最后一张截图之前事件的历史总结。请总结  
9962  
9963  
9964       events **for** future decision-making **and** also propose the most  
9965       这些事件以便未来决策, 并提出最  
9966  
9967  
9968       suitable subtasks to execute **next**, given the overall target task.  
9969       适合执行的子任务, 基于整体目标任务。  
9970  
9971  
9972       Here **is** some helpful information to **help** you do the summarization **and**  
9973       以下是一些有助于你进行总结和  
9974  
9975  
9976       propose the subtask.  
9977       提出子任务的有用信息。  
9978  
9979  
9980       Overall task description:  
9981       整体任务描述:  
9982  
9983  
9984       <\\$task\_description\\$>  
9985  
9986  
9987  
9988       Previous proposed subtask **for** the task:  
9989       之前为该任务提出的子任务:  
9990  
9991  
9992       <\\$subtask\_description\\$>  
9993  
9994  
9995  
9996       Previous reasoning **for** proposing the subtask:  
9997       提出该子任务的前期推理:  
9998  
9999

10000 <\\$subtask\_reasoning\$>  
10001  
10002  
10003  
10004 Image introduction:  
10005 图片介绍:  
10006  
10007  
10008 <\\$image\_introduction\$>  
10009  
10010  
10011  
10012 Last executed action:  
10013 最后执行的操作:  
10014  
10015  
10016 <\\$previous\_action\$>  
10017  
10018  
10019  
10020 Error report for the last executed action:  
10021 最后执行操作的错误报告:  
10022  
10023  
10024 <\\$executing\_action\_error\$>  
10025  
10026  
10027  
10028 Key decision-making reasoning for the last executed action:  
10029 最后执行操作的关键决策推理:  
10030  
10031  
10032 <\\$previous\_reasoning\$>  
10033  
10034  
10035  
10036 Self-reflection for the last executed action:  
10037 最后执行操作的自我反思:  
10038  
10039  
10040 <\\$self\_reflection\_reasoning\$>  
10041  
10042  
10043  
10044 Success\_Detection for the overall task:  
10045 整体任务的成功检测:  
10046  
10047  
10048 <\\$success\_detection\$>  
10049  
10050  
10051  
10052 The following is the summary of history that happened before the last  
10053 以下是最后一张截图之前发生的历史总结  
10054  
10055

10056 screenshot:  
10057 截图：  
10058  
10059  
10060 <\\$previous\_summarization\\$>  
10061  
10062  
10063  
10064 ---  
10065  
10066  
10067  
10068 ---  
10069  
10070  
10071  
10072 History\_summary: Summarize what happened **in** the past experience,  
10073 历史总结：总结过去的经历发生了什么，  
10074  
10075  
10076 especially the last step according to the decision-making  
10077 尤其是根据决策的最后一步  
10078  
10079  
10080 reasoning **and** self-reflection reasoning **for** the last executed  
10081 对最后执行的推理和自我反思推理  
10082  
10083  
10084 action. The summarization needs to be precise, concrete, highly  
10085 行动。摘要需要精准、具体且高度  
10086  
10087  
10088 related to the task, **and** follow the rules below.  
10089 与任务相关，并遵守以下规则。  
10090  
10091  
10092 1. Summarize the tasks **from** the history **and** the current task. What **is**  
10093 1. 总结历史任务和当前任务。内容是什么  
10094  
10095  
10096 the current progress of the task? For example, to **open** a file, you  
10097 任务的当前进展？例如，要打开一个文件，你  
10098  
10099 first need to select the file, then **open** it by clicking somewhere  
10100 首先需要选择文件，然后点击某处打开它  
10101  
10102  
10103  
10104 or using the keyboard. Subtasks may have other pre-requisites.  
10105 或使用键盘。子任务可能有其他先决条件。  
10106  
10107  
10108 2. Record the successful actions **and** organize them into events, step  
10109 2. 记录成功的操作并将其组织成事件，步骤  
10110  
10111

10112 by step.  
10113 逐步进行。  
10114  
10115  
10116 3. Which subtask has been completed? Which subtasks have not?  
10117 3. 哪个子任务已经完成? 哪些子任务尚未完成?  
10118  
10119  
10120 4. Do not forget the information and key events in the previous steps  
10121 4. 不要忘记前面步骤中的信息和关键事件  
10122  
10123  
10124 of the overall task.  
10125 整个任务的。  
10126  
10127  
10128 Subtask\_reasoning: Decide whether the previous subtask is finished and  
10129 子任务推理: 判断前一个子任务是否完成并  
10130  
10131  
10132 whether it is necessary to propose a new subtask. The subtask  
10133 是否有必要提出一个新的子任务。该子任务  
10134  
10135  
10136 should be straightforward, contribute to the target task, and be  
10137 应该直截了当, 有助于目标任务, 并且是  
10138  
10139  
10140 most suitable for the current situation; which should be completed  
10141 最适合当前情况; 应当完成  
10142  
10143  
10144 within a few actions. You should respond with:  
10145 在几步操作内。你应该回复:  
10146  
10147  
10148 1. How to finish the target task? You should analyze it step by step.  
10149 1. 如何完成目标任务? 你应该逐步分析。  
10150  
10151  
10152 2. What is the current progress of the target task according to the  
10153 2. 根据当前情况, 目标任务的进展如何  
10154  
10155  
10156 analysis in question 1? Please do not make any assumptions if  
10157 问题1中的分析? 如果没有明确说明, 请不要做任何假设  
10158  
10159  
10160 needed information is not mentioned previously. You should assume  
10161 所需信息之前未提及。您应当假设  
10162  
10163  
10164 that you are doing the task from scratch. Please strictly follow  
10165 你是从头开始执行该任务的。请严格遵守  
10166  
10167

10168 the description **and** requirements **in** the current overall task.  
10169 当前整体任务中的描述和要求。  
10170  
10171  
10172 3. What **is** the previous subtask? Has the previous subtask finished  
10173 3. 上一个子任务是什么？上一个子任务完成了吗？  
10174  
10175  
10176 according to **self-reflection?** Or **is** it improper **for** the current  
10177 根据自我反思？还是说这对当前来说不合适  
10178  
10179  
10180 situation? If the last subtask already finished **or** now **is** improper  
10181 情况？如果最后一个子任务已经完成或现在不合适  
10182  
10183 , please select a new one. Otherwise you should reuse the last  
10184 , 请选择一个新的。否则您应继续使用上一个。  
10185  
10186  
10187  
10188 subtask .  
10189 子任务。  
10190  
10191  
10192 4. If you propose a new subtask, give the reasons why it **is** more  
10193 4. 如果您提出一个新的子任务，请说明其更优的原因  
10194  
10195  
10196 feasible **in** the current situation **in** the application. Please  
10197 在当前应用环境中是可行的。请  
10198  
10199  
10200 strictly follow the description **and** requirements **in** the current  
10201 严格遵循当前整体任务中的描述和要求。  
10202  
10203  
10204 overall task.  
10205 整体任务。  
10206  
10207  
10208 5. The proposed subtask needs to be precise **and** concrete within one  
10209 5. 所提出的子任务需在一句话内准确具体。  
10210  
10211  
10212 sentence. It should **not** be directly related to **any** skills.  
10213 不应直接涉及任何技能。  
10214  
10215  
10216 You should only respond **in** the **format** described below, **and** you should  
10217 您应仅以以下描述的格式回复，且不得输出评论或其他信息。  
10218  
10219  
10220 **not** output comments **or** other information.  
10221 不得输出评论或其他信息。  
10222  
10223

10224 History\_summary:  
10225 历史摘要：  
10226  
10227  
10228 The summary of past events **is...**  
10229 过去事件的总结是.....  
10230  
10231  
10232 Subtask\_reasoning:  
10233 子任务推理：  
10234  
10235  
10236 1. ...  
10237  
10238  
10239  
10240 2. ...  
10241  
10242  
10243  
10244 ...  
10245  
10246  
10247  
10248 Subtask\_description:  
10249 子任务描述：  
10250  
10251  
10252 The current subtask **is** ...  
10253 当前子任务是.....  
10254  
10255  
10256 ---  
10257  
10258  
10259  
10260 Prompt 46: Feishu: Action Planning prompt.  
10261 提示46：飞书（Feishu）：行动计划提示。  
10262  
10263  
10264 ---  
10265  
10266  
10267  
10268 You an expert helpful AI assistant which follows instructions **and**  
10269 您是一位遵循指令并执行桌面电脑任务的专家级智能助手。  
10270  
10271  
10272 performs desktop computer tasks **as** instructed. You have expert  
10273 您具备飞书（Feishu）PC端的专业知识，能够处理广泛的  
10274  
10275  
10276 knowledge of '**Feishu**' on the PC **and** can handle a wide **range** of  
10277 任务。  
10278  
10279

10280 tasks **in** the application using the keyboard, shortcut keys, **and**  
10281 使用键盘、快捷键执行应用中的任务，和  
10282  
10283  
10284 mouse operations. For each step, you will get one **or** more  
10285 鼠标操作。每一步，您将获得一个或多个  
10286  
10287  
10288 observation images, which are screenshots of the computer screen.  
10289 观察图像，即计算机屏幕的截图。  
10290  
10291 Your advanced capabilities enable you to process **and** interpret  
10292 您的高级能力使您能够处理和解读  
10293  
10294  
10295  
10296 these application screenshots **and** other relevant information **in**  
10297 这些应用程序截图及其他相关信息在  
10298  
10299  
10300 detail.  
10301 详细信息。  
10302  
10303  
10304 Utilizing these insights, you will identify the most suitable **in-**  
10305 利用这些见解，您将确定最合适的内容  
10306  
10307  
10308 application action to take **next**, given the current task. You  
10309 根据当前任务，下一步要采取的应用操作。您  
10310  
10311  
10312 ---  
10313  
10314  
10315  
10316 ---  
10317  
10318  
10319  
10320 control the application **and** can execute actions **from** the available  
10321 控制应用程序并可以执行可用的操作  
10322  
10323  
10324 actions to manipulate its UI. Upon evaluating the provided  
10325 操作其用户界面。评估所提供的  
10326  
10327  
10328 information, you MUST choose the precise actions to perform,  
10329 信息时，您必须选择要执行的具体操作，  
10330  
10331  
10332 considering the applications's present circumstances, and specify  
10333 考虑到应用程序的当前情况，并指定  
10334  
10335

10336       any necessary parameters to execute that action.  
10337       执行该操作所需的任何参数。  
10338  
10339  
10340       Here is some helpful information to help you make the decision.  
10341       这里有一些有助于您做出决定的有用信息。  
10342  
10343  
10344       Overall task description:  
10345       整体任务描述：  
10346  
10347  
10348       <\\$task\_description\\$>  
10349  
10350  
10351  
10352       Subtask description:  
10353       子任务描述：  
10354  
10355  
10356       <\\$subtask\_description\\$>  
10357  
10358  
10359  
10360       Few shots:  
10361       少量截图：  
10362  
10363  
10364       <\\$few\_shots\\$>  
10365  
10366  
10367  
10368       Image introduction:  
10369       图像介绍：  
10370  
10371  
10372       <\\$image\_introduction\\$>  
10373  
10374  
10375  
10376       Current and previous screenshot are the same:  
10377       当前截图与之前截图相同：  
10378  
10379  
10380       <\\$image\_same\_flag\\$>  
10381  
10382  
10383  
10384       Mouse position in the current screenshot is the same as in the  
10385       当前截图中的鼠标位置与  
10386  
10387  
10388       previous screenshot:  
10389       之前截图中的相同：  
10390  
10391

10392 <\\$mouse\_position\_same\_flag\$>  
10393  
10394  
10395  
10396 Description of current screenshot:  
10397 当前截图描述:  
10398  
10399  
10400 <\\$image\_description\$>  
10401  
10402  
10403  
10404 Description of label IDs:  
10405 标签ID描述:  
10406  
10407  
10408 <\\$description\_of\_bounding\_boxes\$>  
10409  
10410  
10411  
10412 Last executed action:  
10413 上一次执行的操作:  
10414  
10415  
10416 <\\$previous\_action\$>  
10417  
10418  
10419  
10420 Key reason for the last action:  
10421 上一次操作的关键原因:  
10422  
10423  
10424 <\\$key\_reason\_of\_last\_action\$>  
10425  
10426  
10427  
10428 Self-reflection for the last executed action:  
10429 对上一次执行操作的自我反思:  
10430  
10431  
10432 <\\$previous\_self\_reflection\_reasoning\$>  
10433  
10434  
10435  
10436 Summarization of recent history:  
10437 近期历史总结:  
10438  
10439  
10440 <\\$previous\_summarization\$>  
10441  
10442  
10443  
10444 Valid action set in Python format to select the next action:  
10445 用于选择下一步操作的Python格式有效操作集:  
10446  
10447

10448 <\\$skill\_library\\$>  
10449  
10450  
10451  
10452 Success detection for overall task:  
10453 整体任务的成功检测：  
10454  
10455  
10456 <\\$success\_detection\\$>  
10457  
10458  
10459  
10460 Based on the above information, you should first analyze the current  
10461 基于以上信息，您应首先分析应用程序的当前  
10462  
10463  
10464 situation of the application and provide the reasoning behind what  
10465 状况，并提供完成任务下一步应采取措施的理由。  
10466  
10467  
10468 should be the next step to complete the task. Then, you should  
10469 然后，您应  
10470  
10471  
10472 output the exact action to be executed in the application.  
10473 输出将在应用程序中执行的具体操作。  
10474  
10475  
10476 Pay attention to all UI items and contents in the image. Before  
10477 注意图像中的所有UI项目和内容。在  
10478  
10479  
10480 changing values or text in the UI, make sure the values in the  
10481 更改UI中的数值或文本之前，确保截图中的数值  
10482  
10483  
10484 screenshot are not already correct for the subtask. DO NOT make  
10485 对于子任务来说尚未正确。不要对布局做出  
10486  
10487  
10488 assumptions about the layout! If the image includes a mouse cursor  
10489 假设！如果图像中包含鼠标光标，  
10490  
10491  
10492 , pay close attention to the coordinates of the pointer tip, not  
10493 请特别注意指针尖端的坐标，而非  
10494  
10495  
10496 the center of the mouse cursor. You should respond with the  
10497 鼠标光标的中心。你应当回复以下  
10498  
10499  
10500 following information, and you MUST answer them one by one.  
10501 信息，且必须逐条回答。  
10502  
10503

10504 ---  
10505  
10506  
10507  
10508 ---  
10509  
10510  
10511  
10512 Decision\_Making\_Reasoning: You should think step by step **and** provide  
10513 决策推理 (Decision\_Making\_Reasoning) : 你应当逐步思考并提供  
10514  
10515  
10516 detailed reasoning to determine the **next** action executed on the  
10517 详细推理, 以确定在当前任务状态下执行的下一步操作。  
10518  
10519  
10520 current state of the task.  
10521 当前任务状态。  
10522  
10523  
10524 1. Does "<\\$success\_detection\\$>" means the overall task was  
10525 1. "<\\$success\_detection\\$>"是否表示整体任务  
10526  
10527  
10528 successful? If successful, ignore questions 2-15. No new action  
10529 成功? 如果成功, 忽略第2至15题。无需采取新操作。  
10530  
10531  
10532 needs to be taken.  
10533 无需采取新操作。  
10534  
10535  
10536 2. You should first describe each item **in** the screen line by line,  
10537 2. 你应先逐行描述屏幕中的每个项目,  
10538  
10539  
10540 from the top left **and** moving right. Is the target item **in** the  
10541 从左上角开始, 向右移动。目标项目是否在  
10542  
10543  
10544 current screen? Which item **is** currently selected?  
10545 当前屏幕? 当前选中了哪个项目?  
10546  
10547  
10548 3. Check whether the UI element you want to operate exists **in** the  
10549 3. 检查您要操作的UI元素是否存在  
10550  
10551  
10552 current screenshot. If **not**, you can choose to move to another part  
10553 当前截图。如果不是, 您可以选择移动到其他部分  
10554  
10555  
10556 of the application, **or** close some recently opened menu item. Also  
10557 应用程序的, 或关闭一些最近打开的菜单项。此外  
10558  
10559

10560 remember that you can use keyboard shortcuts to accomplish  
10561 请记住，您可以使用键盘快捷键来完成  
10562  
10563  
10564 actions, instead of always using the mouse.  
10565 使用快捷键，而不是总是用鼠标。  
10566  
10567  
10568 4. Are there **any** keyboard actions, such as using shortcut keys **or**  
10569 4. 是否有任何键盘操作，例如使用快捷键或  
10570  
10571  
10572 pressing "**enter**", to finish the current step **or** the overall task?  
10573 按"回车"键，完成当前步骤或整个任务？  
10574  
10575  
10576 If so, please specify which one to use. You can always press "  
10577 如果是这样，请指定使用哪一个。您随时可以按"  
10578  
10579  
10580 **enter**" instead of clicking with the mouse, if the button you want  
10581 如果是你想要的按钮，可以按"回车"键代替用鼠标点击  
10582  
10583  
10584 to click on **is active**.  
10585 可点击的是激活状态。  
10586  
10587  
10588 5. If a mouse cursor **is** present **in** the image, describe near which  
10589 5. 如果图像中有鼠标光标，请描述其附近的位置  
10590  
10591  
10592 ID-labeled bounding box **or** unlabelled UI item the cursor's tip is  
10593 光标尖端所在的带ID标签的边界框或未标记的界面元素  
10594  
10595  
10596 located, **not** the center of the cursor.  
10597 位于光标的非中心位置。  
10598  
10599  
10600 6. If the current screenshot **is** the same **as** the previous  
10601 6. 如果当前截图与之前相同  
10602  
10603  
10604 screenshot, DO NOT output the same action **as** **in** the previous step,  
10605 截图，切勿重复上一步的操作，  
10606  
10607  
10608 **as** it was very likely **not** useful.  
10609 因为它很可能没有用处。  
10610  
10611  
10612 7. In the current screenshot, carefully identify the label ID of  
10613 7. 在当前截图中，仔细识别与当前步骤最相关的标签ID  
10614  
10615

the bounding box most relevant to the current step. If there is  
的边界框。如果该边界框内有文本，请提供该文本。如果

text within this bounding box, please provide the text. If there  
没有直接有用的边界框，请提供UI项描述或规范化的**<b0></b0>**坐标。

**is** no directly useful bounding box, provide the UI item  
**description or** normalized \$x,y\$ coordinates.

8. If mouse actions are necessary, specify a bounding box label ID  
8. 如果需要鼠标操作，指定一个边界框标签ID（如果当前截图中显示）作为参数。只有在没有有用的  
标签ID时，才直接生成规范化的**<b0></b0>**坐标。

(**if** shown in the current screenshot) as parameter. Only directly  
generate normalized \$x,y\$ coordinates **if** no useful label ID is  
present.

9. If **not** absolutely sure to be clicking at the right UI item or  
9. 如果不完全确定点击的是正确的UI项或位置，可以先将鼠标移动到该处，查看更多信息。如果确认  
是正确的项，再作为第二步点击它。

location, you can first just move the mouse to it and check for  
more information. If it's the right item, you can click on it in  
as a second step.

10. If there is a dialog or menu opened after the previous action,  
10. 如果在前一步操作后弹出了对话框或菜单，点击其按钮前请注意是否有遗漏的步骤。

pay attention to any missing step before clicking on its buttons.

10670  
10671  
10672 For example, before clicking "Save", make sure a correct file  
10673 例如，在点击"保存"之前，确保在正确的文本框中输入了正确的文件名。  
10674  
10675  
10676 name is typed in the correct text field.  
10677 文件名已输入在正确的文本框中。  
10678  
10679  
10680 11. You should not always use the mouse if you know a keyboard  
10681 11. 如果你知道键盘快捷键或技巧来执行所需操作，不应总是使用鼠标!  
10682  
10683  
10684 shortcut or a skill to perform the desired action!  
10685 快捷键或技巧来执行所需操作!  
10686  
10687  
10688 12. This is the most critical question. Based on the action rules  
10689 12. 这是最关键的问题。基于操作规则和自我反思，  
10690  
10691  
10692 and self-reflection, what should be the most suitable action in  
10693 在有效操作集合中，下一步最合适的操作应该是什么?  
10694  
10695  
10696 the valid action set for the next step? You should analyze the  
10697 你应逐步分析该操作的效果。  
10698  
10699  
10700 effects of the action step by step.  
10701 你应逐步分析该操作的效果。  
10702  
10703  
10704 13. If the previous action is unsuccessful, consider an  
10705 13. 如果前一步操作失败，若可能，考虑替代操作。  
10706  
10707  
10708 alternative action if possible. If there is an alternative action,  
10709 如果存在替代操作，请说明是什么。  
10710  
10711  
10712 please specify what it is. Such as click different label ID or  
10713 例如点击不同的标签ID或使用不同的快捷键。  
10714  
10715  
10716 use different shortcut keys.  
10717 使用不同的快捷键。  
10718  
10719  
10720 14 If you think the next step will be to typing tex, confirm that  
10721 14. 如果你认为下一步是输入文本，确认文本框中已有文本光标，  
10722  
10723  
10724 that there is already a text cursor in it or that the last  
10725 或上一步操作是在合适的输入区域点击。

10726  
10727  
10728     executed action was a click at the appropriate **input** area. If  
10729     如果两者都不满足，必须点击相应的输入区域。  
10730  
10731  
10732     neither **is** true, it **is** mandatory to click on the corresponding  
10733     对应的输入区域。  
10734  
10735  
10736     **input** box before proceeding **with** typing.  
10737     在继续输入之前，请先点击输入框。  
10738  
10739  
10740     15. If you need to interact **with** an UI item that has no bounding  
10741     15. 如果您需要与没有边界的UI元素进行交互  
10742  
10743  
10744     box label ID, you can use its x, y coordinates. Use normalized  
10745     框标签ID，您可以使用其x、y坐标。使用归一化  
10746  
10747  
10748     values **from** 0 **to** 1 .  
10749     数值范围从0到1。  
10750  
10751  
10752     Actions: The best action, **or** short sequence of actions without gaps,  
10753     动作：最佳动作，或无间断的短动作序列，  
10754  
10755  
10756     to execute **next** to progress **in** achieving the goal. Pay attention  
10757     执行下一步以推进目标的实现。请注意  
10758  
10759  
10760     to the names of the available skills **and** to the previous skills  
10761     到可用技能的名称和之前的技能  
10762  
10763  
10764     already executed, **if any**. Pay special attention to the coordinates  
10765     已执行（如有）。特别注意坐标  
10766  
10767  
10768     of **any** action that needs them. Do **not** make assumptions about the  
10769     任何需要它们的操作。不要对其做出假设。  
10770  
10771  
10772     location of UI elements **or** their coordinates, analyse **in** detail  
10773     界面元素的位置或其坐标，进行详细分析  
10774  
10775  
10776     **any** provided images. You should also pay more attention to the  
10777     任何提供的图像。你还应更加关注  
10778  
10779  
10780     following action rules:  
10781     以下操作规则：

10782  
10783  
10784 1. If "<\\$success\_detection\\$>" means the overall task was  
10785 1. 如果"<\\$success\_detection\\$>"表示整体任务已完成  
10786  
10787  
10788 successful or equal to "True", then output action MUST be empty  
10789 如果成功或等于"True", 则输出操作必须为空  
10790  
10791  
10792 like ". Be careful to check the task was really successful.  
10793 像"。请务必确认任务确实成功完成。  
10794  
10795  
10796 2. You should output actions in Python code format and specify any  
10797 2. 你应以Python代码格式输出操作，并说明任何  
10798  
10799  
10800 necessary parameters to execute that action. Only use function  
10801 执行该操作所需的参数。仅使用函数  
10802  
10803  
10804 names and argument names exactly as shown in the valid actions et.  
10805 名称和参数名称，必须与有效操作集中的完全一致。  
10806  
10807  
10808 If a function has parameters, you should also include their names  
10809 如果函数有参数，也应包含参数名称  
10810  
10811  
10812 and decide their values, like "press\_shift (duration=1)". If it  
10813 并决定其值，例如 "press\_shift(duration=1)"。如果  
10814  
10815  
10816 does not have a parameter, just output the action, like "  
10817 没有参数，则只需输出该操作，如 "  
10818  
10819  
10820 release\_mouse\_buttons().  
10821 release\_mouse\_buttons().  
10822  
10823  
10824 3. Before typing text, ensure that the last executed action  
10825 3. 在输入文本前，确保上一个执行的操作  
10826  
10827  
10828 involved clicking on the relevant input box. If the last action  
10829 是点击了相关的输入框。如果上一个操作  
10830  
10831  
10832 was not a click on this input box, the required action MUST be to  
10833 不是点击该输入框，则必须先执行点击对应输入框的操作，  
10834  
10835  
10836 click on the corresponding input box before proceeding.  
10837 然后再继续。

10838  
10839  
10840     4. Given the current situation **and** task, you should only choose  
10841     4. 根据当前情况和任务，应仅从有效操作集中  
10842  
10843  
10844     the most suitable action **from** the valid action **set**. If values **in**  
10845     选择最合适的操作。如果屏幕上的值已正确，  
10846  
10847  
10848     the screen are already correct, no need **for** a new action.  
10849     则无需执行新的操作。  
10850  
10851  
10852     5. When you decide to perform a mouse action, **if** there **is** bounding  
10853     5. 当决定执行鼠标操作时，如果当前截图中有边界框，  
10854  
10855  
10856     box **in** the current screenshot, you **MUST** choose skill  
10857     必须选择技能  
10858  
10859  
10860     click\_on\_label(label\_id, mouse\_button).  
10861     click\_on\_label(label\_id, mouse\_button).  
10862  
10863  
10864     6. When you perform a mouse action, always select the target UI  
10865     6. 当你执行鼠标操作时，总是选择距离上一次操作的UI元素最近的目标UI  
10866  
10867  
10868     element closest to the UI element of the previous action **for**  
10869     元素进行操作。  
10870  
10871  
10872     operation.  
10873  
10874  
10875  
10876     7. When you decide to operate on a file, such **as** downloading it,  
10877     7. 当你决定对文件进行操作，例如下载时，  
10878  
10879  
10880     please pay attention to the path **and** name of the current file.  
10881     请注意当前文件的路径和名称。  
10882  
10883  
10884     8. If upon **self-reflection** you think the target coordinates were  
10885     8. 如果经过自我反思你认为目标坐标存在问题，  
10886  
10887  
10888     an issue, you **MUST** pay close attention to choosing new coordinates  
10889     你必须特别注意选择与之前不同或不太相似的新坐标。  
10890  
10891  
10892     that are **not** the same **or** too similar to the previous ones.  
10893

10894  
10895  
10896 9. If upon **self**-reflection you think the last action was  
10897 9. 如果经过自我反思你认为上一次操作在当前状态下不可用,  
10898  
10899  
10900 unavailable at the current state, you SHOULD **try** to take another  
10901 你应尝试采取其他操作以启用所需的操作。  
10902  
10903  
10904 action to **try** to enable the desired action.  
10905  
10906  
10907  
10908 10. If you leave the application incorrectly, you can go back to  
10909 10. 如果你错误地离开了应用程序, 可以直接使用 `go_back_to_target_application()` 返回,  
10910  
10911  
10912 it directly using `go_back_to_target_application()`. No need to use  
10913 无需使用鼠标。  
10914  
10915  
10916 the mouse.  
10917  
10918  
10919  
10920 You should only respond **in** the **format** described below. In your  
10921 你应仅以以下描述的格式进行回应。在你选择的操作理由中,  
10922  
10923  
10924 reasoning **for** the chosen actions, also describe which item you  
10925 还要描述你选择了哪个项目。  
10926  
10927 decided to interact **with** and why. DO NOT change the title of each  
10928 决定与谁互动以及原因。不要更改每个  
10929  
10930  
10931 item. You should **not** output other comments **or** information besides  
10932 条目的标题。除以下格式外, 不要输出其他评论或信息  
10933  
10934  
10935  
10936 the **format** below:  
10937 :  
10938  
10939  
10940 Decision\_Making\_Reasoning:  
10941 决策推理:  
10942  
10943  
10944 1. ...  
10945  
10946  
10947  
10948 2. ...  
10949

10950  
10951  
10952 3. ...  
10953  
10954  
10955  
10956 Actions:  
10957 操作:  
10958  
10959  
10960 cccpython  
10961 cccpython  
10962  
10963  
10964 action ( \${args1} = x, \${args2} = y\$ )  
10965 操作 ( \${args1} = x, \${args2} = y\$ )  
10966  
10967  
10968 ccc  
10969 ccc  
10970  
10971  
10972 Key\_reason\_of\_last\_action:  
10973 上次操作的关键原因:  
10974  
10975  
10976 ...  
10977  
10978  
10979  
10980 ----  
10981