



Scratch

Nâng cao

💡 Phiên bản 3.0



Chương I: Dự án ôn tập cơ bản

Bài 1: Trò chơi hứng bóng

Bài 2: Trò chơi tung bóng

Chương II: Ứng dụng AI

Bài 3: Dùng AI hỗ trợ hướng dẫn

Bài 4: Dùng AI hỗ trợ sửa lỗi

Chương III: Nhóm lệnh Bút vẽ

Bài 5: Thêm khối lệnh bút vẽ

Bài 6: Khối lệnh vẽ cơ bản

Bài 7: Lệnh hiệu chỉnh nét vẽ

Chương IV: Khối lệnh lưu trữ

Bài 8: Biến số toàn bộ nhân vật

Bài 9: Biến số cho một nhân vật

Chương V Dự án tổng hợp

Bài 10: Đồng hồ kim

Bài 11: Trò chơi hứng trái cây

Bài 12: Trò chơi đánh bóng bàn

Bài 13: Trò chơi bắn chim

Bài 14: Trò chơi đoán số

Bài 15: Trò chơi đua xe

Chương 1: Dự án ôn tập cơ bản

Bài 1 Trò chơi hứng bóng

Bài 2 Trò chơi tung bóng



Bài 1. Trò chơi hứng bóng

Mục tiêu bài học

- Ôn tập các nội dung cơ bản
- Thực hành các dự án luyện tập

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Các quả bóng liên tục rơi từ trên xuống, công việc của bạn là hứng các quả bóng này, nếu bạn hứng trượt, game over.

2) Các Bước Thực Hiện

Bước 1: Thêm một phong Nền

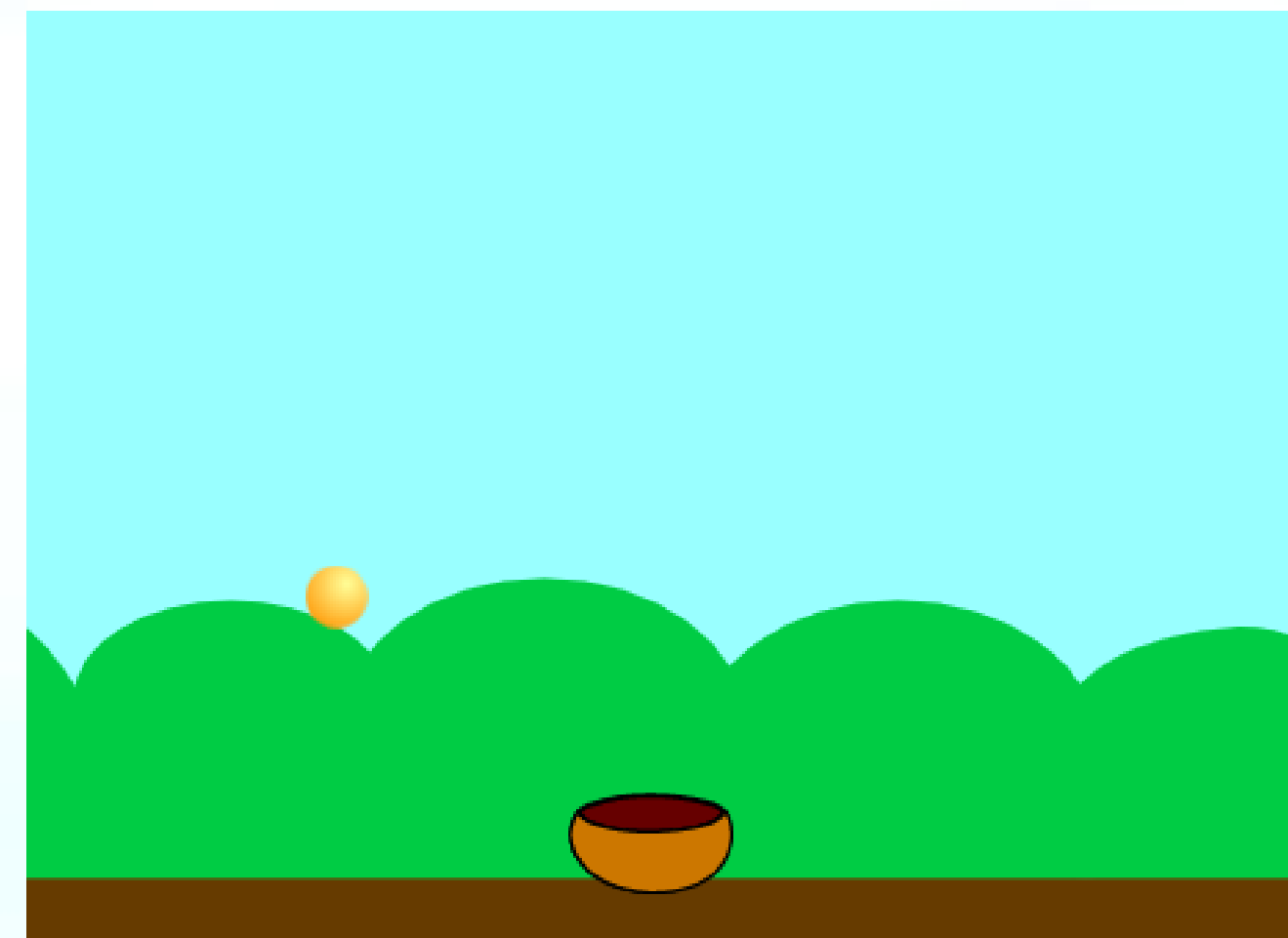
Bước 2: Nhân vật: - Quả bóng
- Cái tô

Bước 3: Lập trình cho nhân vật

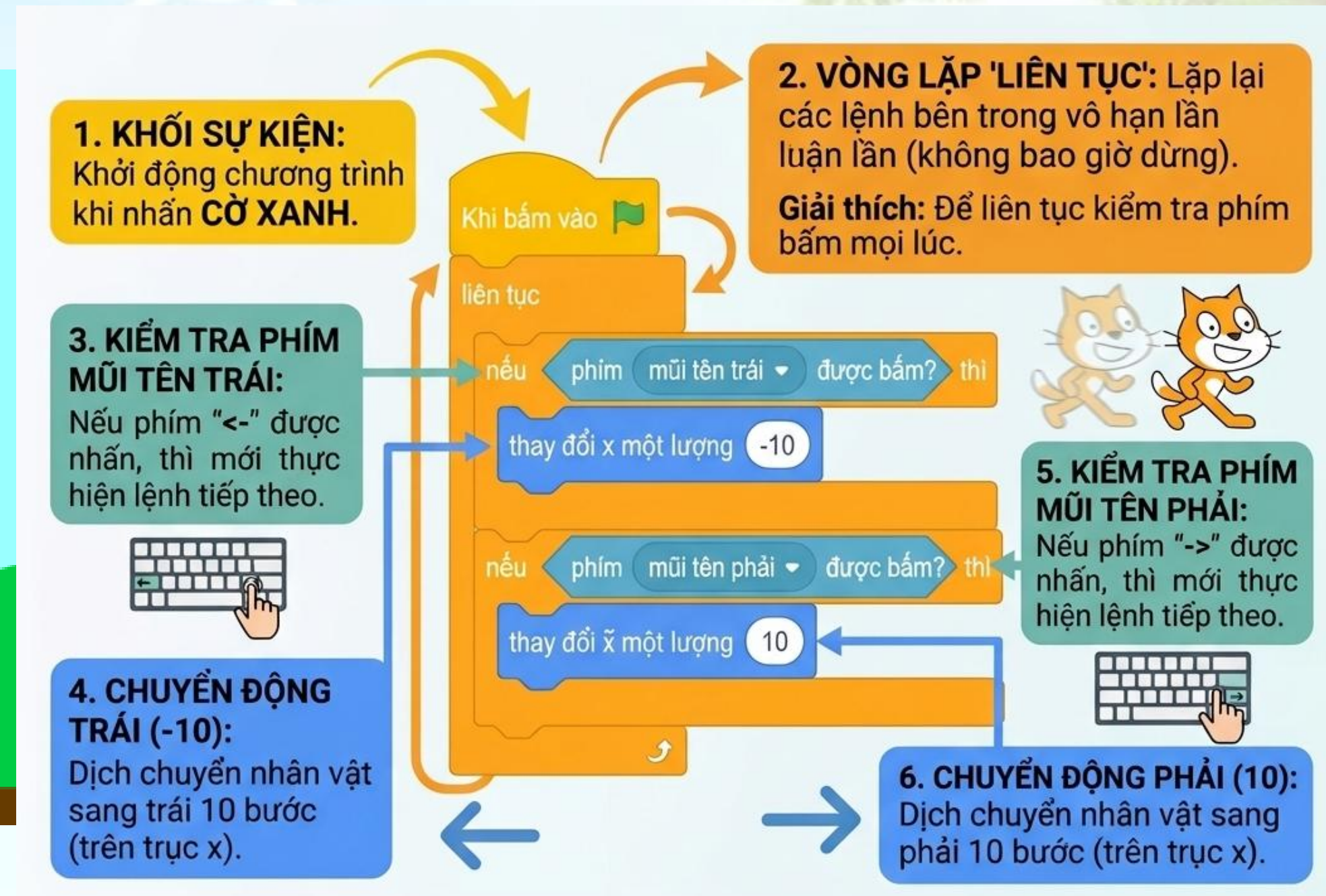
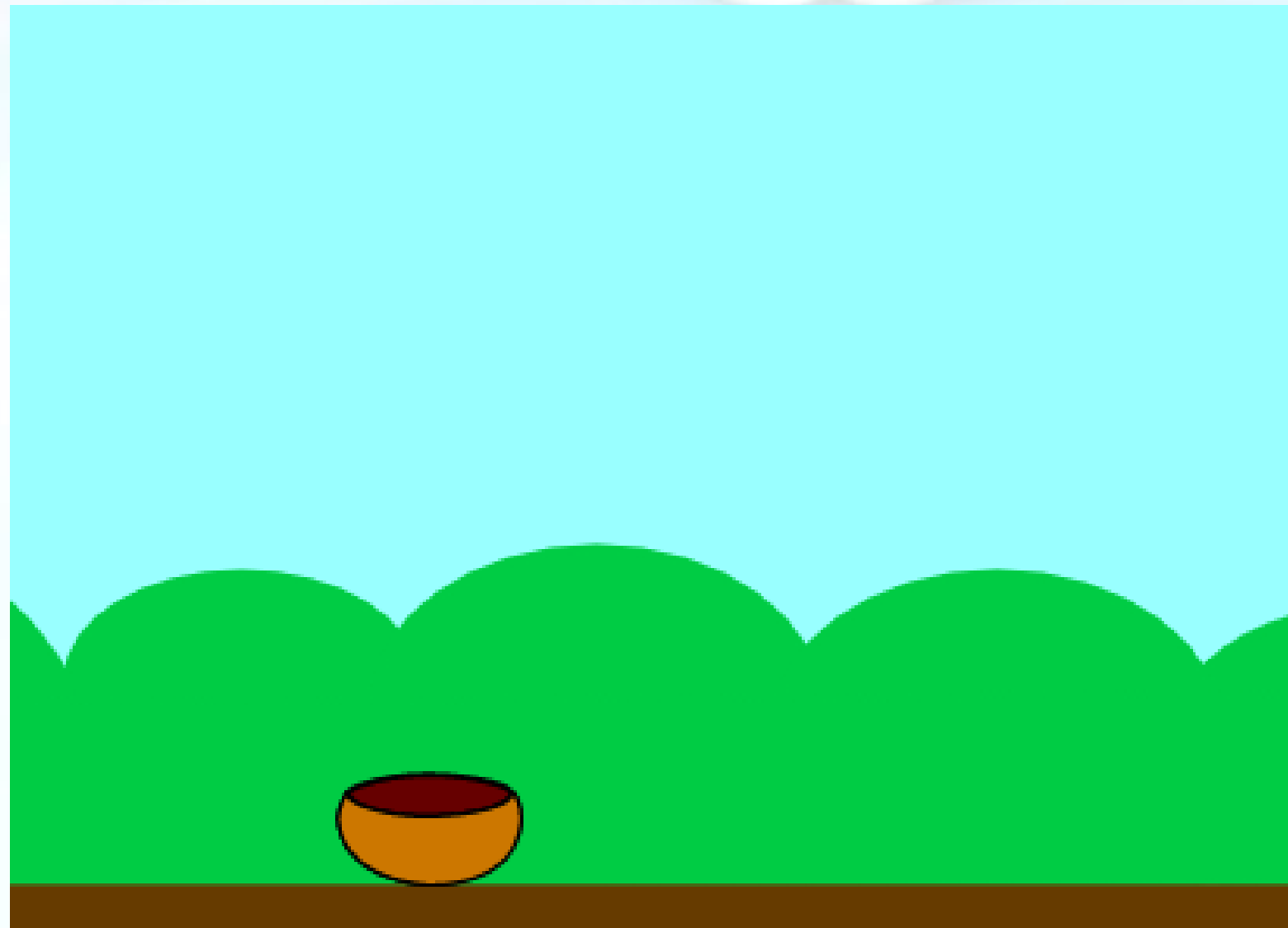
- 3.1 Cái tô: di chuyển trái – phải bằng phím Mũi tên
- 3.2 Quả bóng: liên tục tạo các bản sao và rơi xuống.

Nếu chạm phải tô, quả bóng sẽ biến mất.

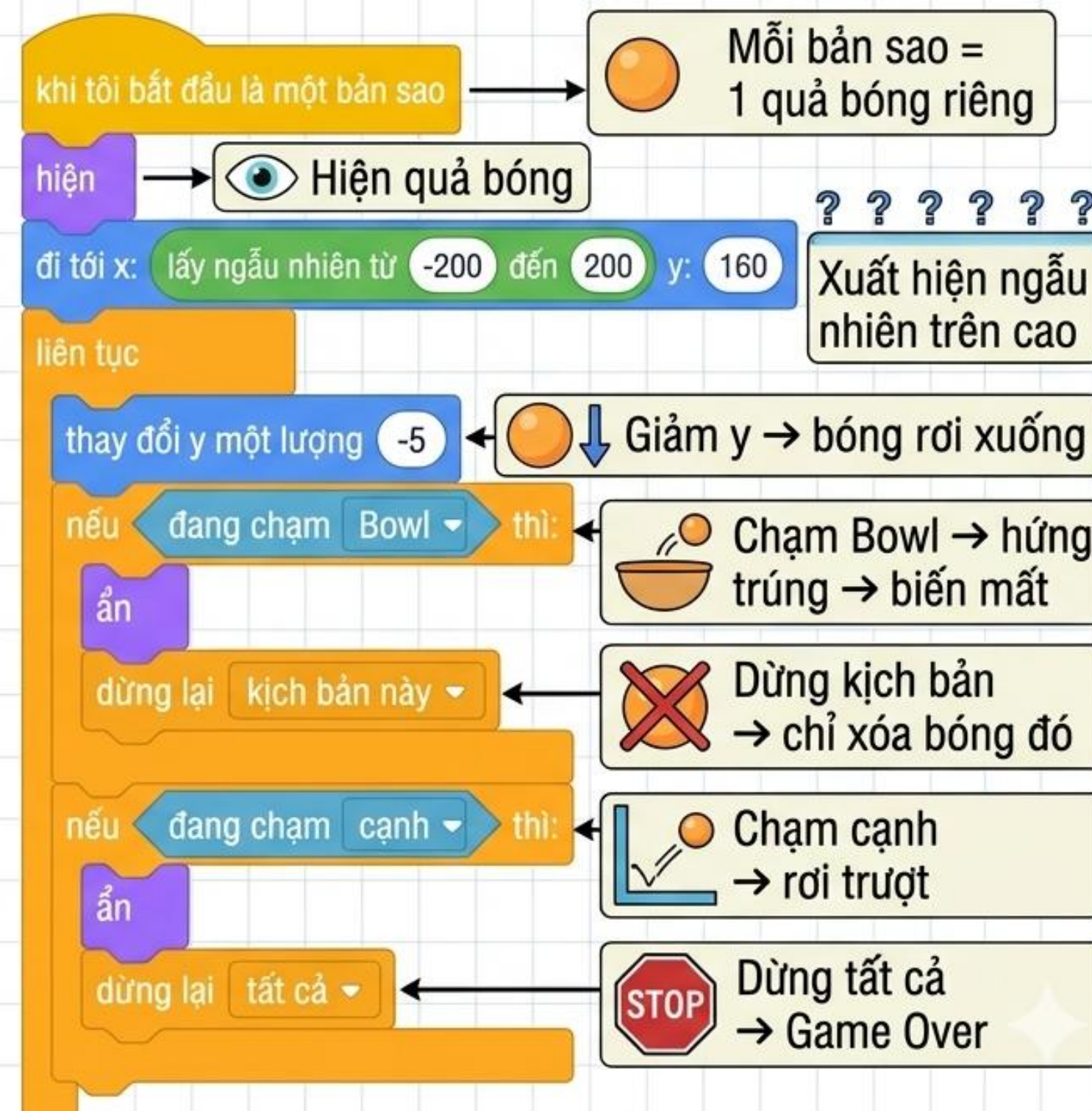
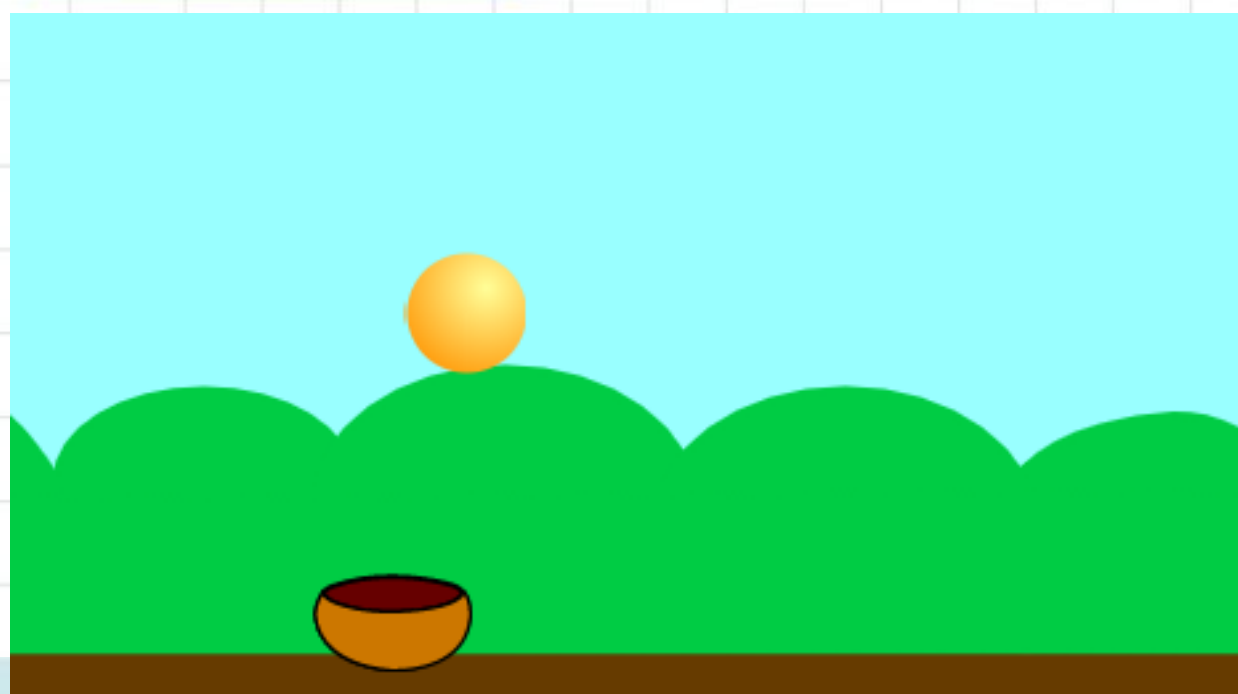
Nếu chạm vào cạnh dưới, game over



Bước 3.1 Lập trình điều khiển cái tô



Bước 3.1 Lập trình cho quả bóng rơi



Bài 2. Trò chơi tưng bóng

Mục tiêu bài học

- Ôn tập các nội dung cơ bản
- Thực hành xây dựng trò chơi tưng bóng

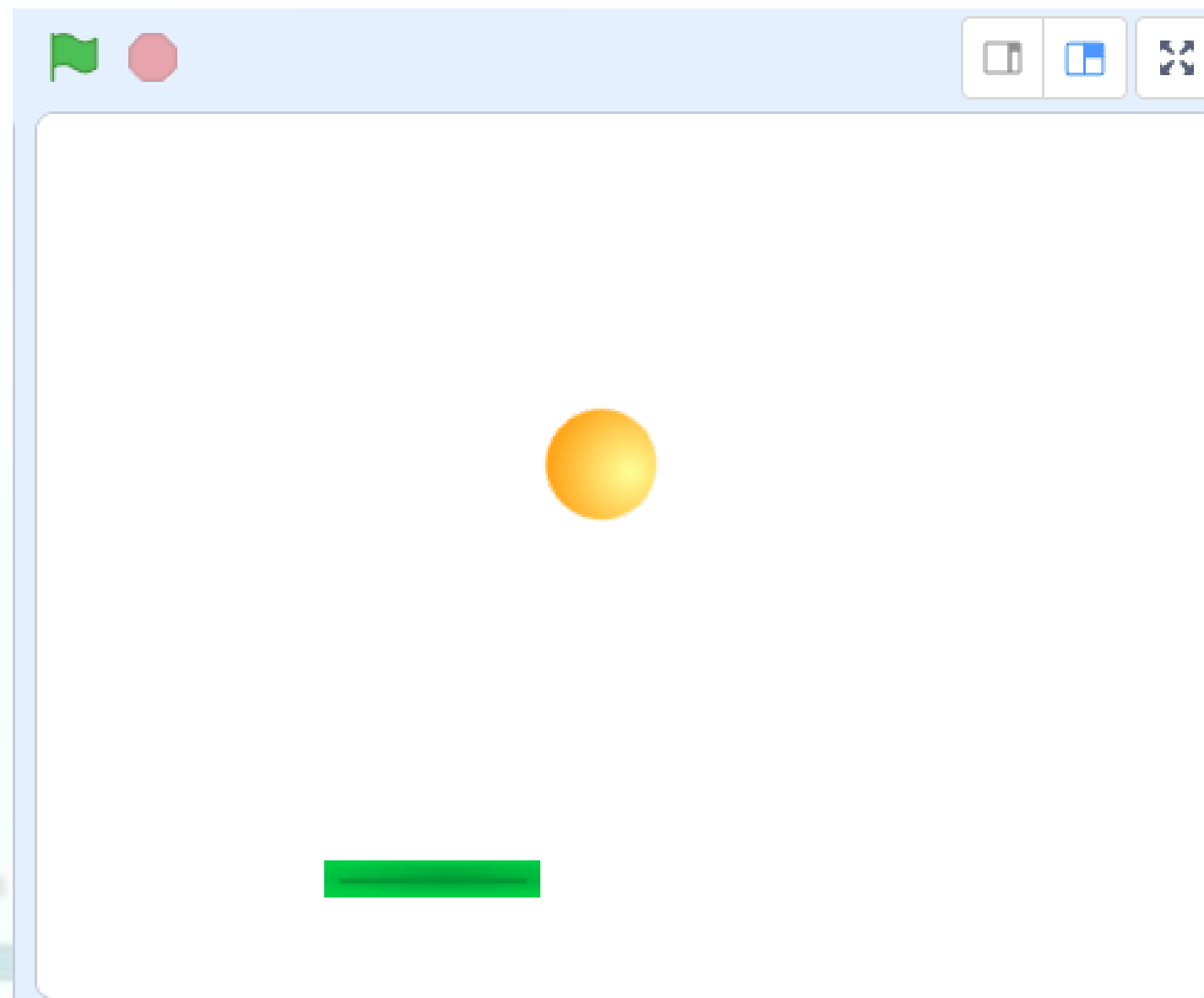
Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Quả bóng sẽ rơi tự do từ trên trần xuống. Nếu gặp cạnh sân khấu sẽ tự động bật lại và chuyển động liên tục. Nếu quả bóng chạm thanh ngang, nó sẽ bật lên 1 góc ngẫu nhiên từ -45 đến 45 độ và chuyển động ngược lên trên. Người dùng điều khiển dùng các phím phải, trái để điều khiển quả bóng sao cho bóng không bị rơi xuống phía dưới. Nếu bóng bị rơi, trò chơi dừng lại

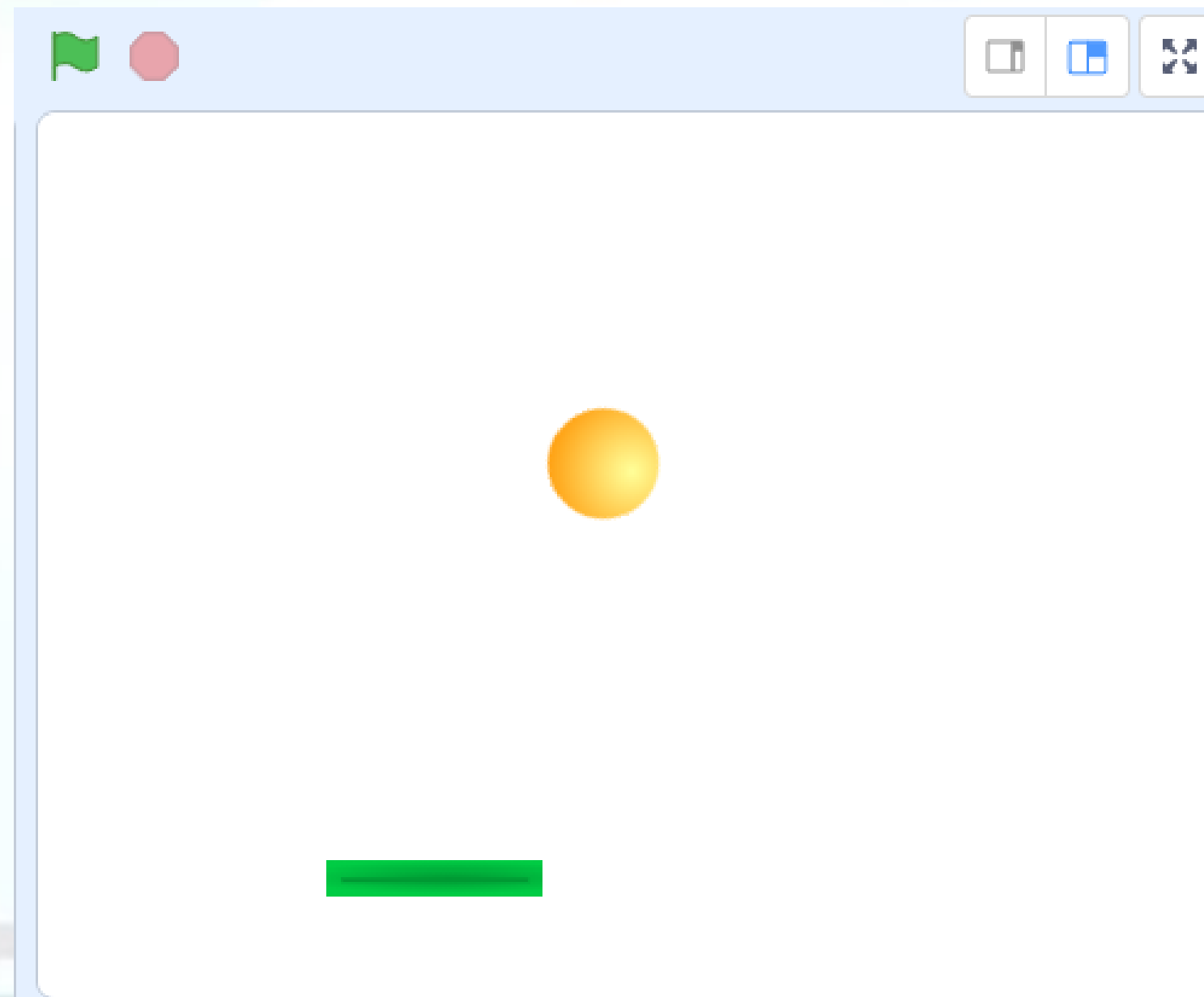


2) Các Bước Thực Hiện

Bước 1: Lên ý tưởng cho trò chơi - Trái bóng: Liên tục di chuyển trên màn hình. Bật lại khi chạm cạnh - Thanh chắn: Di chuyển sang trái – phải bằng phím mũi tên, ngăn cho trái bóng không rơi xuống.

Bước 2: Tạo nhân vật

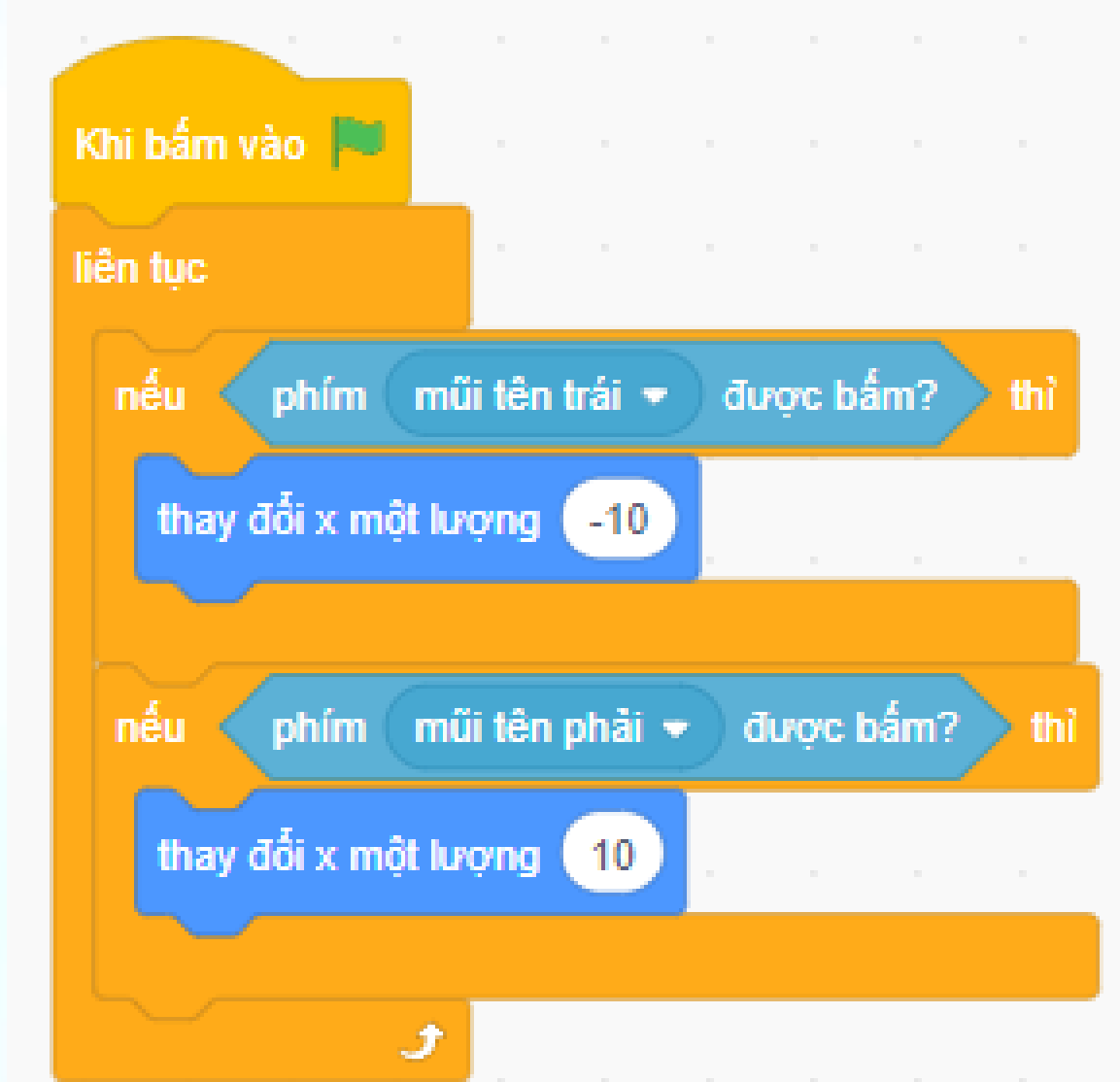
- + Quả bóng
- + Thanh chắn



Bước 3: Lập trình cho thanh thước

Điều khiển sang trái phải bằng phím mũi tên

BƯỚC 1 Khi bấm vào 	★ Giải thích: 👉 Chương trình bắt đầu chạy khi người dùng nhấn vào cờ xanh. 
BƯỚC 2 liên tục 	★ Giải thích: 👉 Vòng lặp vô hạn, giúp kiểm tra phím bấm liên tục, đảm bảo nhân vật phản hồi ngay lập tức. 
BƯỚC 3 nếu  mũi tên trái ▾ được bấm? thì thay đổi x một lượng -10	★ Giải thích: 👉 Nếu người chơi nhấn phím mũi tên trái, nhân vật sẽ di chuyển sang trái (giảm tọa độ X). 👉 Giá trị -10 làm nhân vật di chuyển nhanh hơn. 
BƯỚC 4 nếu  mũi tên phải ▾ được bấm? thì thay đổi x một lượng 10	★ Giải thích: 👉 Nếu nhấn phím mũi tên phải, nhân vật sẽ di chuyển sang phải (tăng tọa độ X). 



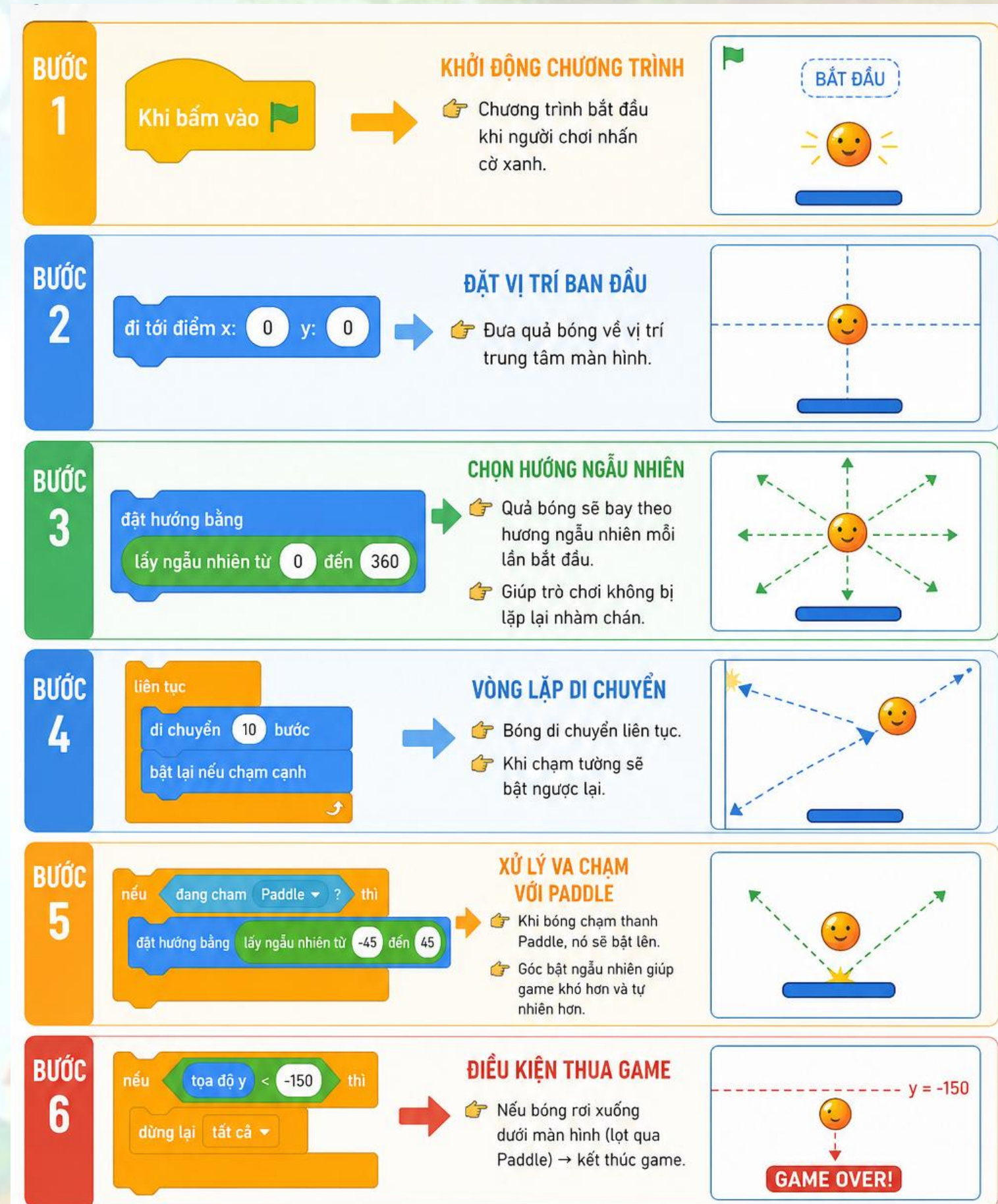
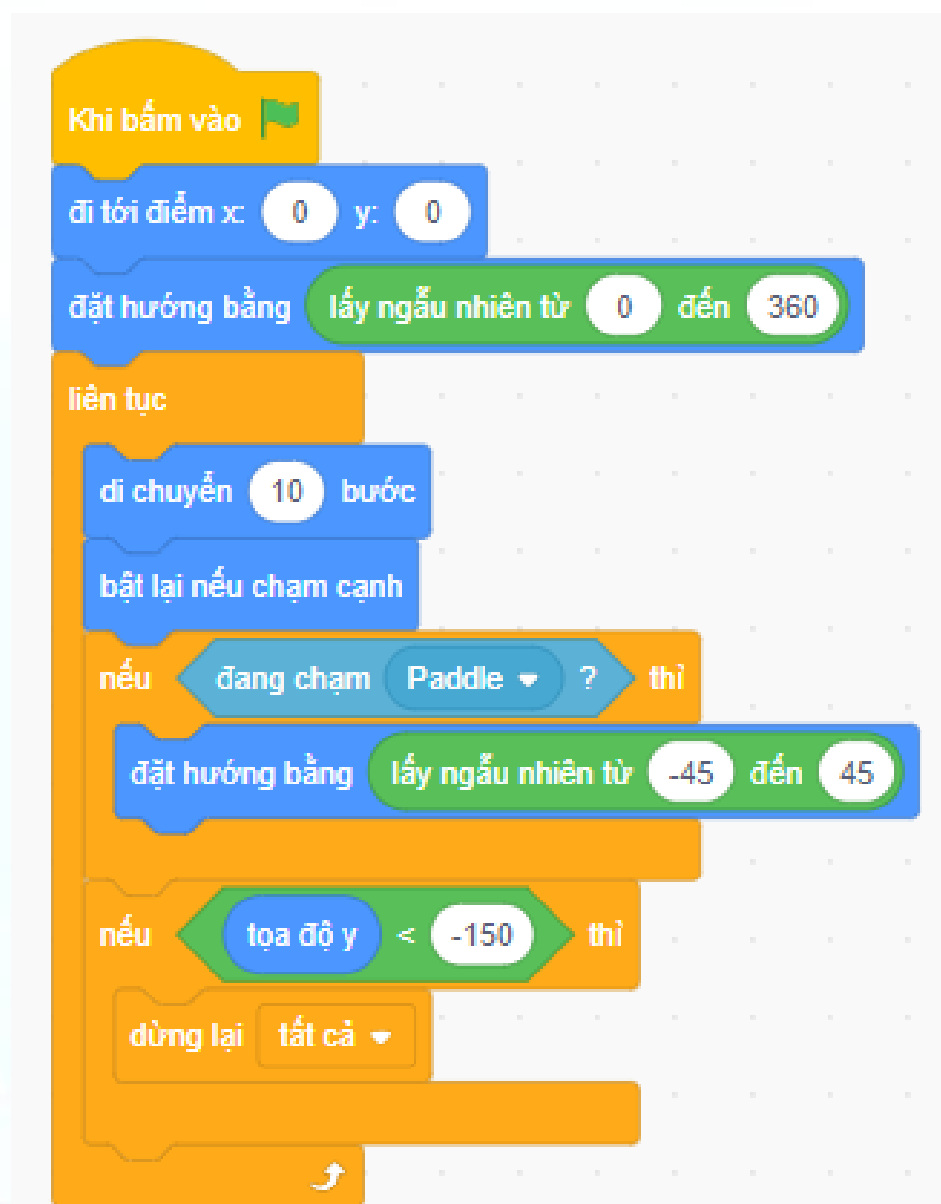
```

    Khi bấm vào [cờ xanh]
    liên tục
    nếu [phím mũi tên trái ▾ được bấm?] thì
        thay đổi x một lượng -10
    nếu [phím mũi tên phải ▾ được bấm?] thì
        thay đổi x một lượng 10
    
```



Bước 4: Lập trình cho Quả bóng

Quả bóng xuất hiện giữa màn hình và di chuyển với hướng ngẫu nhiên, bật lại nếu chạm cạnh. Nếu chạm cây thước sẽ nảy lên với góc ngẫu nhiên từ -45 đến 45 độ. Nếu thanh thước đỡ trượt, bóng rơi. Game over



Chương 2: Ứng dụng AI

Bài 3 Ứng dụng AI hướng dẫn lập trình

Bài 4 Ứng dụng AI sửa lỗi chương trình



Bài 3. Ứng dụng AI hướng dẫn lập trình

Mục tiêu bài học

- Biết cách sử dụng công cụ AI
- Ứng dụng AI để hướng dẫn lập trình

Nội dung nghiên cứu

- 1) Công cụ AI thường dùng
- 2) Ra lệnh AI hướng dẫn lập trình



1) Công cụ AI thường dùng

Công cụ trí tuệ nhân tạo có thể giúp ích rất nhiều cho chúng ta trong việc lập trình, ví dụ như: hỗ trợ xây dựng thuật toán, viết chương trình, sửa lỗi...

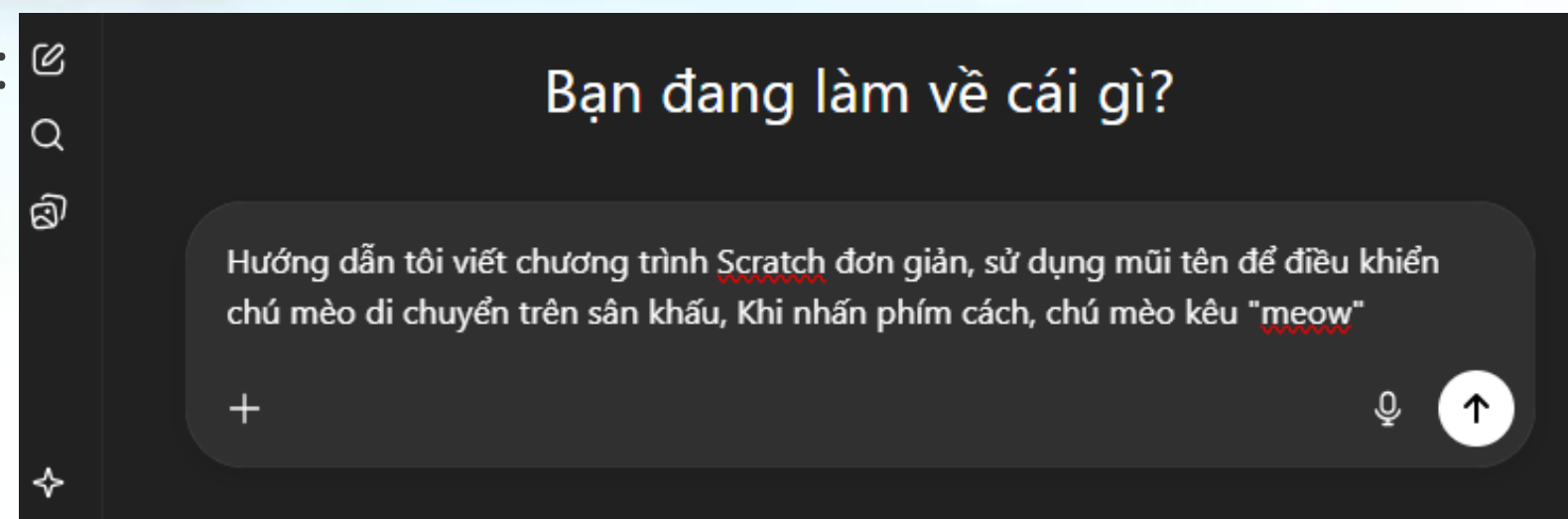
Dưới đây là một số công cụ AI phổ biến thường dùng



2) Ra lệnh cho AI hướng dẫn lập trình

Bước 1: Mở công cụ AI (Ví dụ: Chat GPT)

Bước 2: Ra lệnh cho AI. Vd:



Bước 3: Thực hiện theo hướng dẫn để hoàn thành chương trình

🌟 Các bước thực hiện trong Scratch

Bước 1: Mở Scratch

- Vào <https://scratch.mit.edu> hoặc mở Scratch Offline.
- Tạo một dự án mới.

Bước 2: Chọn nhân vật

- Giữ nguyên nhân vật mặc định là con mèo (Sprite: Cat).
- Nếu lỡ xóa, nhấn Chọn nhân vật → Chọn mèo từ thư viện.

Bước 3: Viết chương trình điều khiển di chuyển

1. Chọn nhân vật mèo.
2. Vào nhóm lệnh Sự kiện (Events) và Chuyển động (Motion).
3. Thêm các đoạn mã sau:



Bài 4. Ứng dụng AI sửa lỗi chương trình

Mục tiêu bài học

- Biết cách sử dụng công cụ AI
- Ứng dụng AI để sửa lỗi chương trình

Nội dung nghiên cứu

Ra lệnh AI Yêu cầu sửa lỗi chương trình

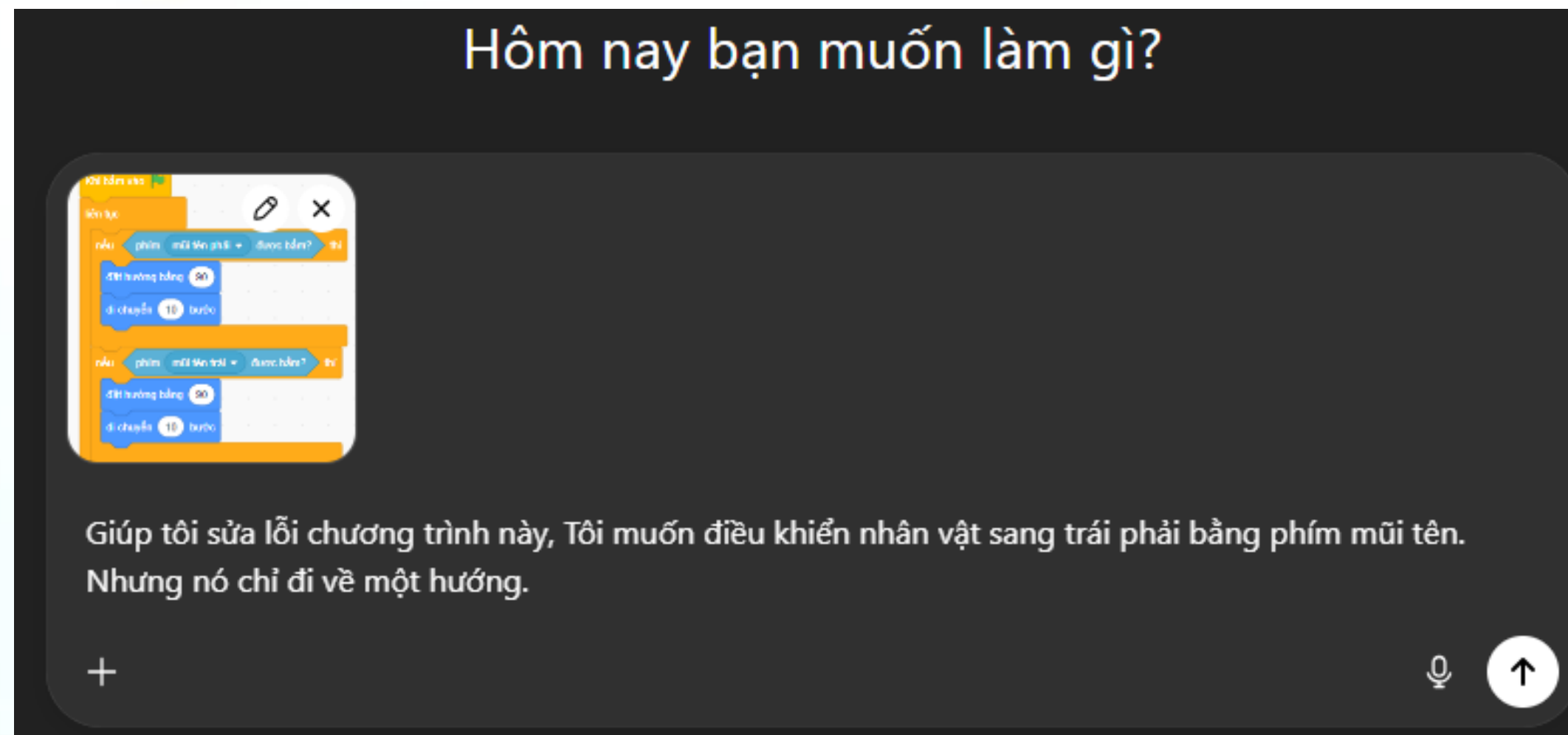


Ra lệnh cho AI hướng dẫn lập trình

Bước 1: Mở công cụ AI (Ví dụ: Chat GPT)

Bước 2: Chụp ảnh khối lệnh chương trình bị lỗi

Bước 3: Ra lệnh cho AI yêu cầu sửa lỗi một chương trình



Bước 4: Thực hiện theo hướng dẫn để sửa lại chương trình

Bạn làm rất đúng rồi, chỉ sai **một chi tiết nhỏ** nên nhân vật chỉ di chuyển về **một hướng (phải)**.

👉 Lý do là: trong cả hai khối **nếu**, bạn đều **đặt hướng bằng 90 độ**, tức là luôn quay sang phải.

✅ **Cách sửa đúng:**

Thay đổi hướng ở phần **phím mũi tên trái** thành **-90 độ**, như sau:

🔧 **Mã hoàn chỉnh:**

CSS

📋 Sao chép mã

Khi bấm vào lá cờ xanh
liên tục

nếu <phím **[mũi tên phải v]** được bấm?> thì

đặt hướng bằng **(90)**

di chuyển **(10)** bước

nếu <phím **[mũi tên trái v]** được bấm?> thì

đặt hướng bằng **(-90)**

di chuyển **(10)** bước



Chương 3: Nhóm lệnh bút vẽ

Bài 5 Thêm khối lệnh vẽ

Bài 6 Các khối lệnh vẽ cơ bản

Bài 7 Các khối lệnh hiệu chỉnh nét vẽ



Bài 5. Thêm khối lệnh vẽ

Mục tiêu bài học

- Biết cách thêm các khối lệnh từ phần mở rộng
- Thêm khối lệnh bút vẽ vào Scratch

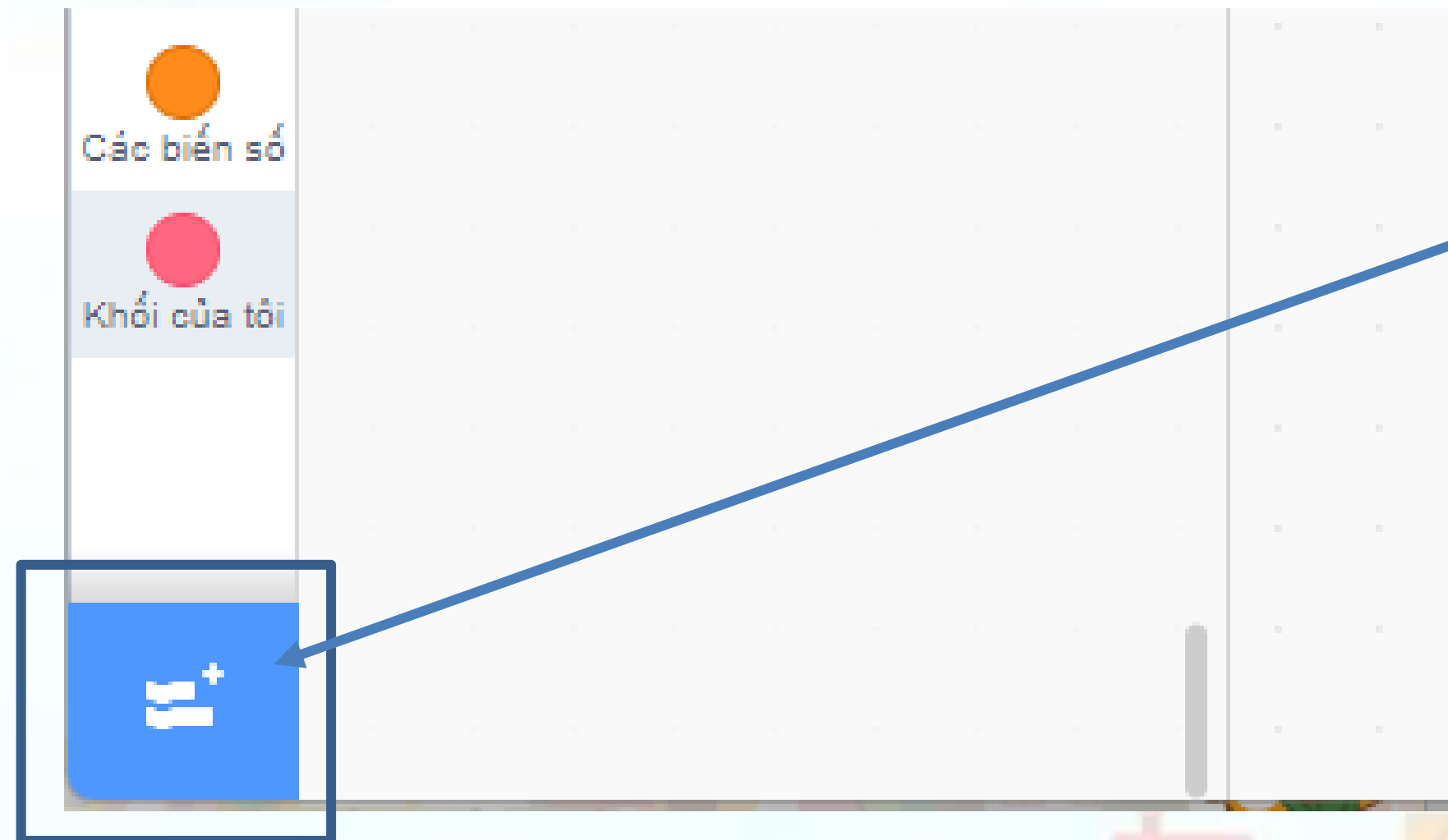
Nội dung nghiên cứu

- 1) Phần mở rộng
- 2) Thêm khối lệnh vẽ vào Scratch



1) Phần mở rộng

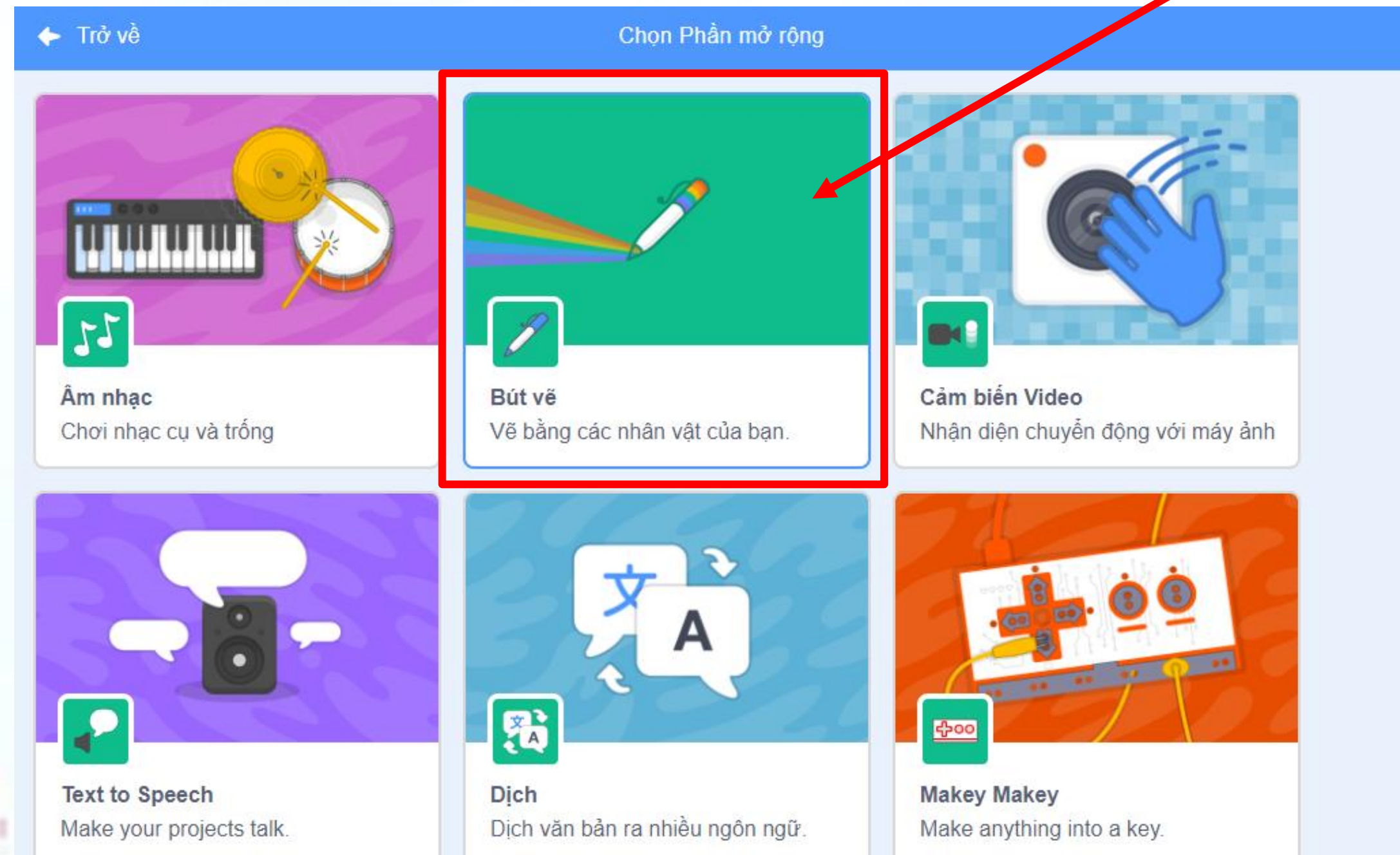
Trong Scrath, một số tiện ích không có sẵn, vì vậy, ta cần thêm phần mở rộng để sử dụng những tiện ích đó.



2) Thêm nhóm Bút vẽ

Bước 1: Chọn nút thêm phần mở rộng

Bước 2: Chọn công cụ bút vẽ



Bài 6. Khối lệnh vẽ cơ bản

Mục tiêu bài học

- Hiểu được ý nghĩa các khối lệnh vẽ cơ bản
- Sử dụng khối lệnh vẽ để vẽ các hình cơ bản

Nội dung nghiên cứu

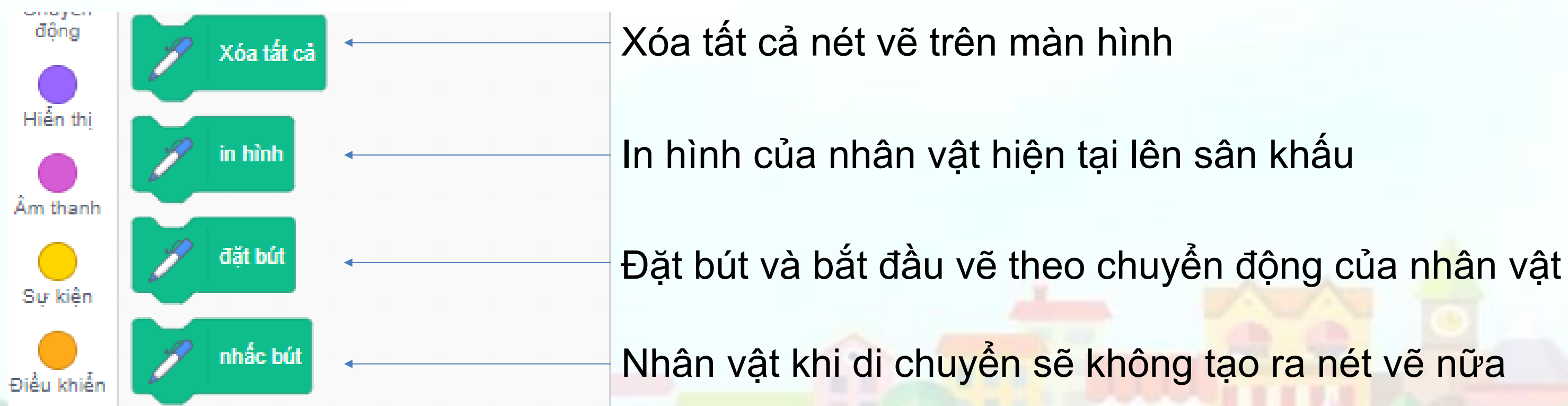
- 1) Ý nghĩa
- 2) Các khối lệnh cơ bản



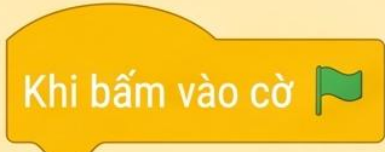
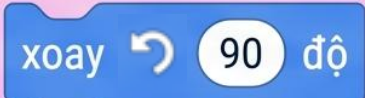
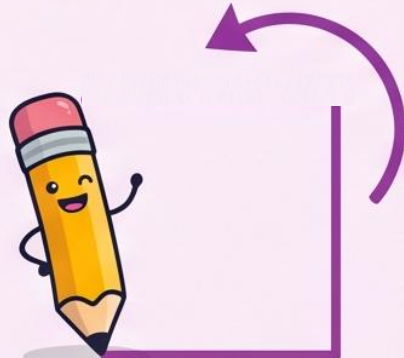
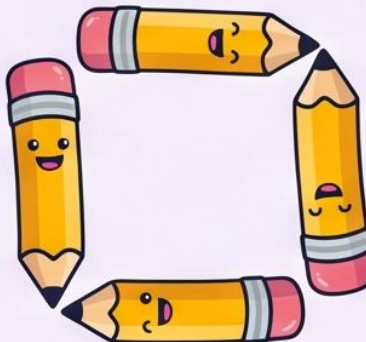
1) Ý nghĩa

Các khối lệnh vẽ cơ bản trong Scratch giúp điều khiển quá trình tạo hình trên sân khấu. Người dùng có thể đặt bút để bắt đầu vẽ theo chuyển động của nhân vật, nhắc bút để ngừng vẽ, in hình của nhân vật lên sân khấu, và xóa tất cả nét vẽ khi cần làm mới màn hình. Nhờ đó, việc vẽ và điều khiển nét vẽ trở nên linh hoạt và dễ dàng hơn.

2) Các lệnh vẽ cơ bản



3) Ví dụ: Vẽ hình vuông cạnh 100 bước

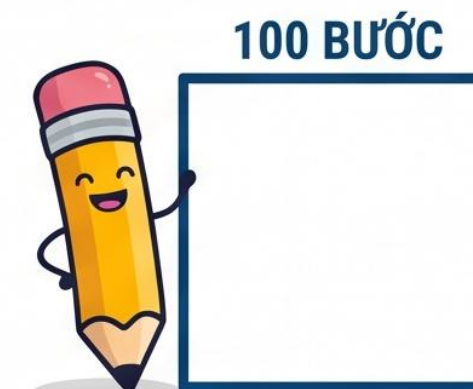
Bước 1 Bắt đầu	Bước 2 Chuẩn bị bút	Bước 3 Vẽ cạnh	Bước 4 Quay góc	Bước 5 Lặp hoàn chỉnh
				
👉 Chương trình bắt đầu khi nhấn cờ xanh. 	👉 Đưa bút về vị trí bắt đầu và sẵn sàng vẽ. 	👉 Vẽ một cạnh dài 100 bước. 	👉 Quay trái 90° để tạo góc vuông. 	👉 Lặp lại 4 lần để tạo đủ 4 cạnh của hình vuông. 

Code hoàn chỉnh

```

Khi bấm vào 🚩
  xóa tất cả
  nhấc bút
  đi tới x: 0 y: 0
  đặt hướng 90 độ
  hạ bút
  lặp lại 4 lần:
    di chuyển 100 bước
    xoay 90 độ

```



Bài 7. Khối lệnh hiệu chỉnh nét vẽ

Mục tiêu bài học

- Hiểu được ý nghĩa các khối lệnh hiệu chỉnh
- Sử dụng khối lệnh hiệu chỉnh để thay đổi độ dày, màu sắc vẽ

Nội dung nghiên cứu

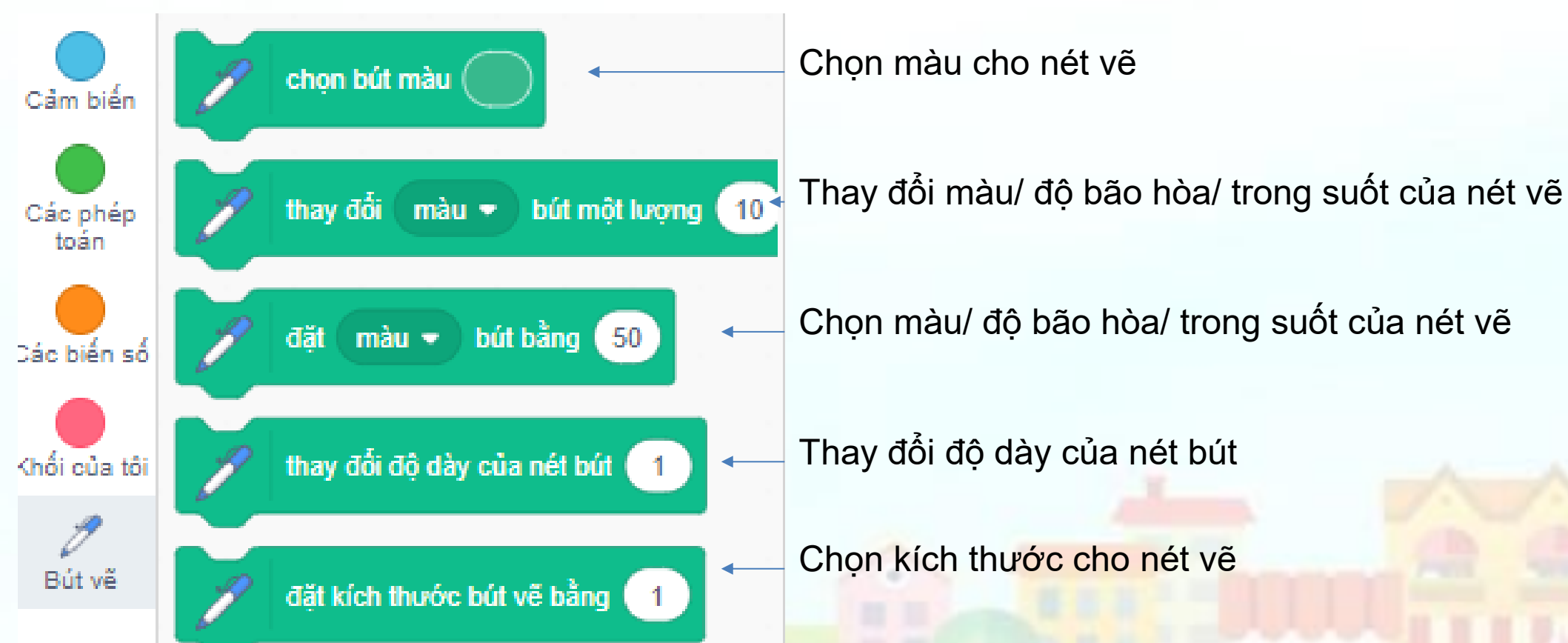
- 1) Các khối lệnh hiệu chỉnh nét vẽ
- 2) Bài tập thực hành



1) Ý nghĩa

Các khối lệnh hiệu chỉnh bút vẽ trong Scratch giúp thay đổi đặc tính của nét vẽ. Ta có thể chọn màu bút, thay đổi độ dày hoặc kích thước của nét bút. Nhờ đó, các hình vẽ trở nên sinh động và đa dạng hơn.

2) Các khối lệnh hiệu chỉnh bút vẽ



The image shows the Scratch drawing tool menu with the following blocks and their corresponding labels:

- Chọn bút màu** (Choose pen color) - Chọn màu cho nét vẽ
- thay đổi màu/ độ bão hòa/ trong suốt của nét vẽ** (Change color/ saturation/ transparency of the stroke) - Thay đổi màu/ độ bão hòa/ trong suốt của nét vẽ
- đặt màu/ độ bão hòa/ trong suốt của nét vẽ** (Set color/ saturation/ transparency of the stroke) - Chọn màu/ độ bão hòa/ trong suốt của nét vẽ
- thay đổi độ dày của nét bút** (Change stroke thickness) - Thay đổi độ dày của nét bút
- đặt kích thước bút vẽ bằng** (Set drawing tool size to) - Chọn kích thước cho nét vẽ

3) Ví dụ: Vẽ hình vuông cạnh 100 bước, nét vẽ màu đỏ, dày 5.

- 1 Bắt đầu chương trình**
Khi nhấn lá cờ xanh, cây bút chì bắt đầu chạy chương trình.
- 2 Xóa nét vẽ cũ**
Xóa tất cả hình đã vẽ trước đó để bắt đầu bản vẽ mới.
- 3 Nhấc bút**
Cây bút chì di chuyển mà chưa vẽ đường.
- 4 Về vị trí bắt đầu**
Đưa cây bút chì về giữa sân khấu Scratch tại tọa độ x:0 y:0.
- 5 Đặt hướng**
Cho cây bút chì quay sang phải để chuẩn bị vẽ.
- 6 Chọn màu bút**
Đổi màu nét vẽ thành màu đỏ.
- 7 Đặt độ dày nét vẽ**
Thiết lập độ dày của nét vẽ là 5 để hình vuông rõ hơn.
- 8 Đặt bút xuống**
Khi di chuyển, cây bút chì sẽ bắt đầu vẽ.
- 9 Vẽ hình vuông**
di chuyển 100 bước → vẽ 1 cạnh
xoay 90 độ → tạo góc vuông
Lặp lại 4 lần để tạo đủ 4 cạnh của hình vuông.

Hình vuông có 4 cạnh bằng nhau và 4 góc vuông nên cần lặp lại 4 lần.



Chương 4: Khối lệnh lưu trữ

Bài 8 Biến số toàn bộ nhân vật

Bài 9 Biến số chỉ một nhân vật



Bài 8. Biến số toàn bộ nhân vật

Mục tiêu bài học

- Hiểu được ý nghĩa của biến toàn bộ nhân vật
- Ứng dụng biến số để lập trình tính điểm, đếm số, tính toán...

Nội dung nghiên cứu

- 1) Ý nghĩa
- 2) Các khối lệnh lưu trữ



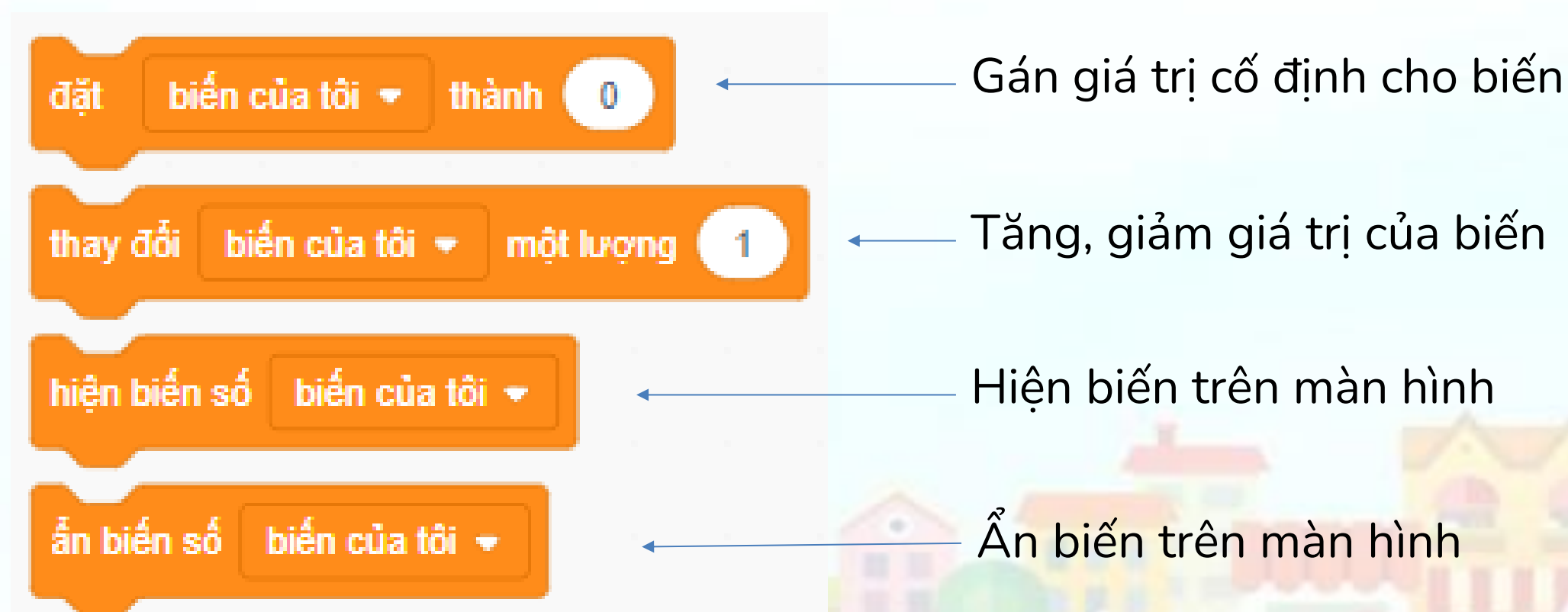
1) Ý nghĩa

- Trong Scratch, **biến số** là ô lưu trữ thông tin, giúp chương trình ghi nhớ và thay đổi dữ liệu như điểm số hoặc tên người chơi.

- **Biến số toàn bộ nhân vật** là biến có thể được sử dụng và thay đổi bởi mọi nhân vật trong dự án.

Nghĩa là giá trị của biến này được chia sẻ chung giữa tất cả các nhân vật, giúp chúng có thể trao đổi thông tin hoặc phối hợp hành động trong chương trình.

2) Các khối lệnh lưu trữ



The image shows four Scratch variable blocks stacked vertically, each with an annotation pointing to it from the right:

- Block 1:** "đặt biến của tôi ▼ thành 0" (Set my variable to 0). Annotation: "Gán giá trị cố định cho biến" (Assign a fixed value to the variable).
- Block 2:** "thay đổi biến của tôi ▼ một lượng 1" (Change my variable by 1). Annotation: "Tăng, giảm giá trị của biến" (Increase, decrease the value of the variable).
- Block 3:** "hiện biến số biến của tôi ▼" (Show variable my variable ▼). Annotation: "Hiện biến trên màn hình" (Show variable on the screen).
- Block 4:** "ẩn biến số biến của tôi ▼" (Hide variable my variable ▼). Annotation: "Ẩn biến trên màn hình" (Hide variable on the screen).



3) Ví dụ: Viết chương trình đồng hồ đếm giây. Khi bấm vào nút Start, sẽ bắt đầu đếm giây.



Bài 9. Biến số chỉ một nhân vật

Mục tiêu bài học

- Hiểu được ý nghĩa của biến một nhân vật
- Ứng dụng biến một nhân vật

Nội dung nghiên cứu

- 1) Ý nghĩa
- 2) Sự khác nhau giữa biến số toàn nhân vật và một nhân vật



1) Ý nghĩa

- Trong Scratch, **biến số chỉ một nhân vật** nghĩa là chỉ nhân vật đó mới nhìn thấy và sử dụng được ô lưu trữ này. Các nhân vật khác không thể xem hay thay đổi giá trị bên trong, giúp mỗi nhân vật có thông tin riêng của mình.trình.

2) So sánh và ứng dụng

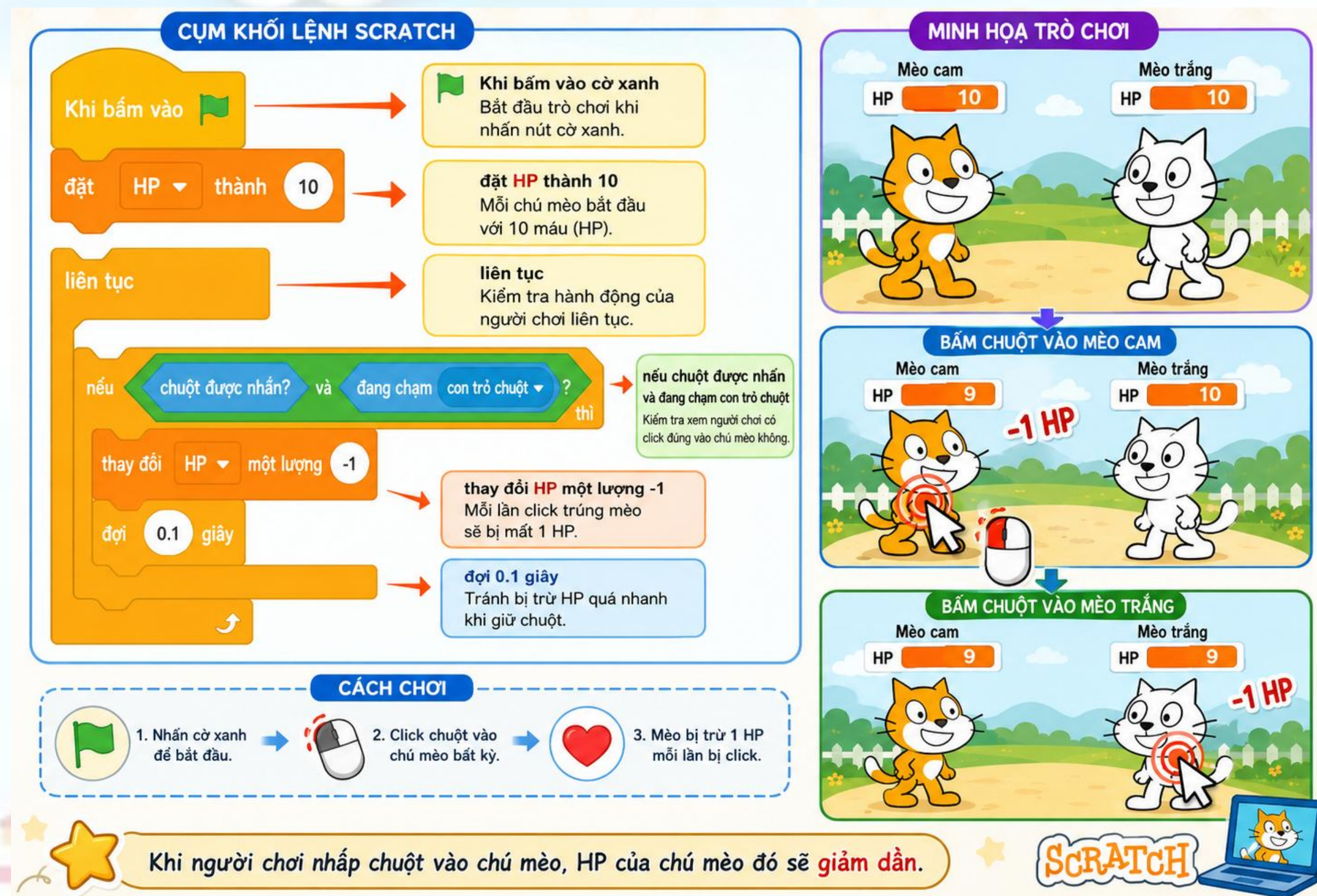
- **Biến số chỉ một nhân vật** là ô lưu trữ riêng của nhân vật đó, các nhân vật khác không thấy và không dùng được. Còn **biến số cho toàn bộ nhân vật** là ô lưu trữ dùng chung, mọi nhân vật đều có thể xem và thay đổi.

- Ứng dụng:

- + Biến riêng: dùng cho thông tin cá nhân của từng nhân vật, như vị trí, tốc độ, máu,...
- + Biến chung: dùng cho thông tin chung của trò chơi, như điểm số, thời gian, lượt chơi,...



3) Ví dụ: Viết chương trình với 2 chú mèo, 2 chú mèo đều có 10 điểm HP, khi người chơi bấm chuột vào chú mèo, chú mèo đó sẽ bị trừ 1 HP



Chương 5: Dự án tổng hợp

- Bài 10 Đồng hồ kim
- Bài 11 Trò chơi hứng bóng
- Bài 12 Trò chơi tung bóng
- Bài 13 Trò chơi bắn chim
- Bài 14 Đường lên đỉnh Olympia
- Bài 15 Trò chơi đua xe



Bài 10. Đồng hồ kim

Mục tiêu bài học

- Biết sử dụng các lệnh thời gian thực trong Scratch
- Hiểu cách xoay kim giờ, kim phút, kim giây theo thời gian
- Tạo được đồng hồ kim hoạt động tự động trên sân khấu

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các Bước thực hiện



1) Tổng quan chương trình

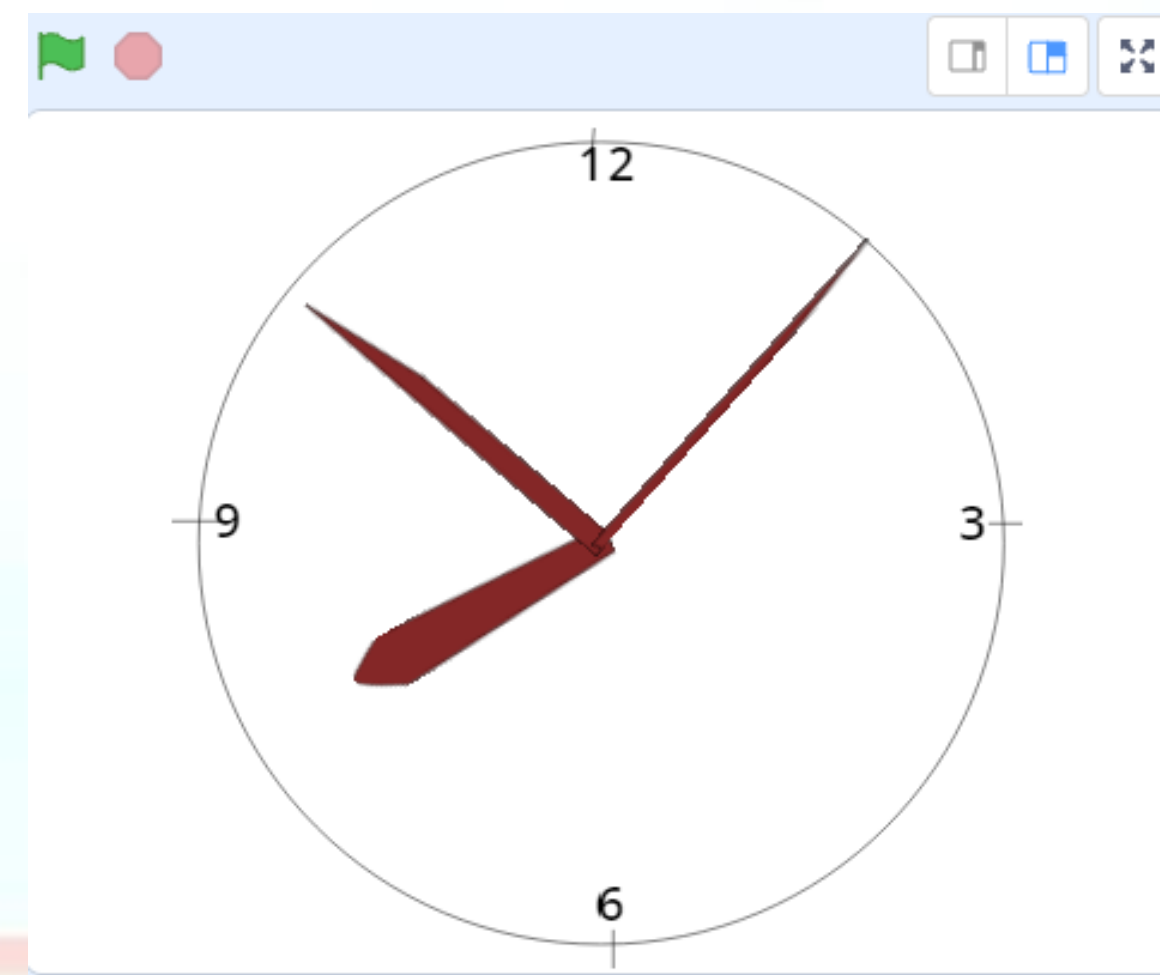
Dự án giúp học viên tạo một chiếc đồng hồ kim hiển thị thời gian thực bằng Scratch. Mỗi kim (giờ, phút, giây) được lập trình xoay theo giá trị thời gian hiện tại. Qua đó, học viên rèn luyện kỹ năng sử dụng lệnh thời gian, tính toán góc quay và điều khiển chuyển động nhân vật, tạo nên một sản phẩm sinh động và hữu ích.

2) Các Bước Thực Hiện

Bước 1: Phong nền chiếc mặt đồng hồ kim. Có thể tự vẽ hoặc tải từ internet.

Bước 2: Nhân vật: - Kim giờ, Kim phút, Kim giây

Bước 3: Lập trình cho nhân vật



Bước 3: Lập trình cho nhân vật

1 KIM GIÂY

Khi bấm vào → Bắt đầu chạy đồng hồ.

liên tục → Cập nhật vị trí kim giây liên tục.

đặt hướng bằng $360 / 60 * \text{giây} \downarrow$ **hiện tại**

→ Xoay kim giây đúng với thời gian hiện tại.

$360 / 60 * \text{giây hiện tại}$

Một vòng tròn có 360° .
Kim giây quay đủ 1 vòng sau 60 giây.



2 KIM PHÚT

Khi bấm vào → Bắt đầu chạy đồng hồ.

liên tục → Cập nhật vị trí kim phút liên tục.

đặt hướng bằng $360 / 60 * \text{phút} \downarrow$ **hiện tại**

→ Xoay kim phút theo phút hiện tại.

$360 / 60 * \text{phút hiện tại}$

Kim phút quay đủ 1 vòng sau 60 phút.

3 KIM GIỜ

Khi bấm vào → Bắt đầu chạy đồng hồ.

liên tục → Cập nhật vị trí kim giờ liên tục.

đặt hướng bằng $360 / 12 * \text{giờ} \downarrow$ **hiện tại**

→ Xoay kim giờ theo giờ hiện tại.

$360 / 12 * \text{giờ hiện tại}$

Kim giờ quay đủ 1 vòng sau 12 giờ.



Ba kim của đồng hồ sẽ tự động xoay theo giờ, phút và giây hiện tại của máy tính.

Bài 11. Trò chơi hứng trái cây

Mục tiêu bài học

- Biết điều khiển nhân vật di chuyển sang trái, phải để hứng vật rơi.
- Hiểu cách phát hiện va chạm giữa nhân vật và trái cây.
- Biết tính điểm khi hứng đúng và kết thúc trò chơi khi bỏ lỡ.
- Rèn luyện kỹ năng lập trình điều khiển, tính điểm và tạo chuyển động rơi trong Scratch.

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Các trái cây liên tục rơi từ trên xuống, công việc của bạn là hứng các trái cây này, mỗi quả được cộng 1 điểm. nếu bạn hứng trượt 3 lần, game over.

2) Các Bước Thực Hiện

Bước 1: Thêm một phong Nền

Bước 2: Nhân vật: -Trái cây
-Cái tô

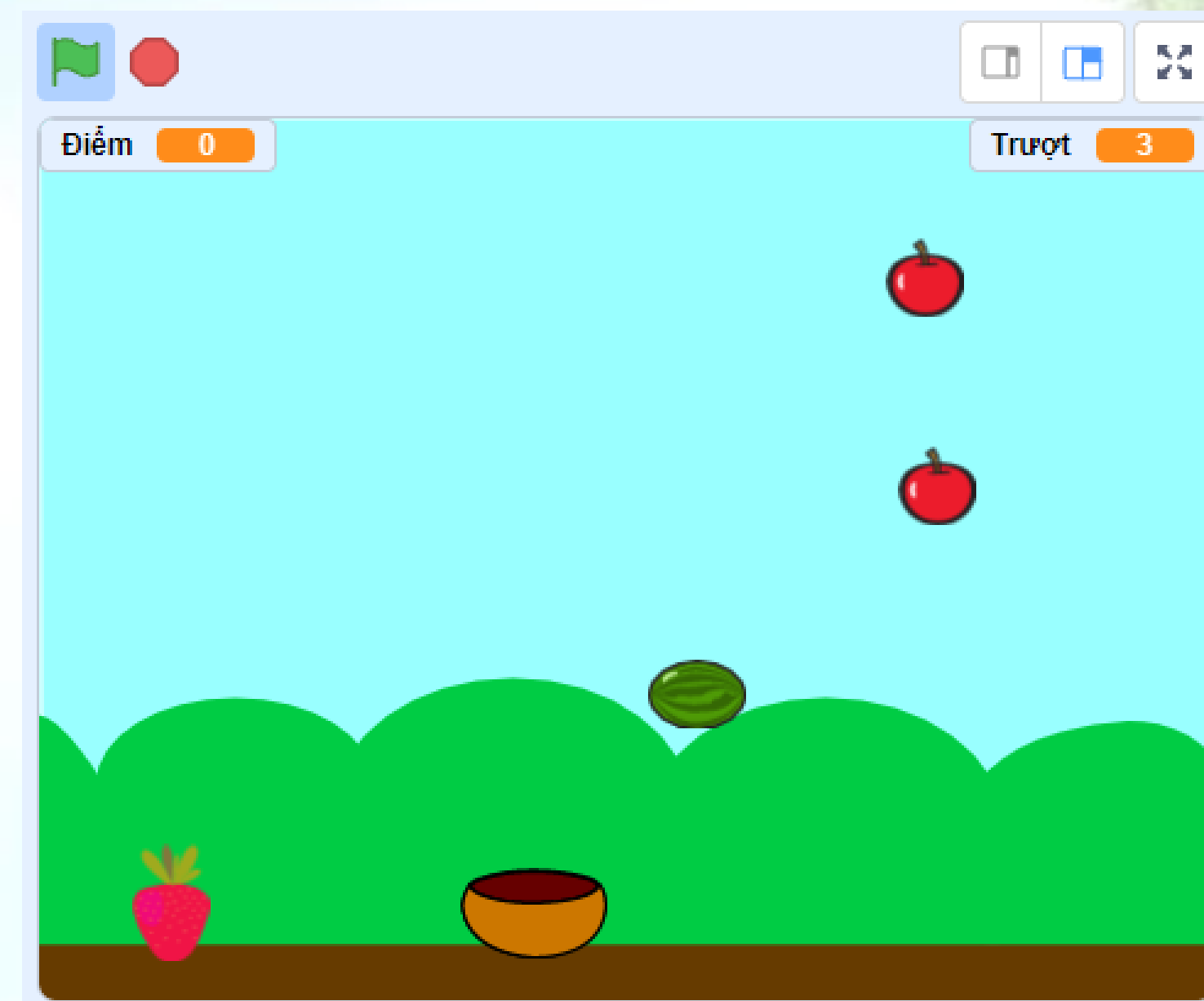
Bước 3: Lập trình cho nhân vật

- 3.1 Cái tô: di chuyển trái – phải bằng phím Mũi tên
- 3.2 Trái cây: liên tục tạo các bản sao và rơi xuống.

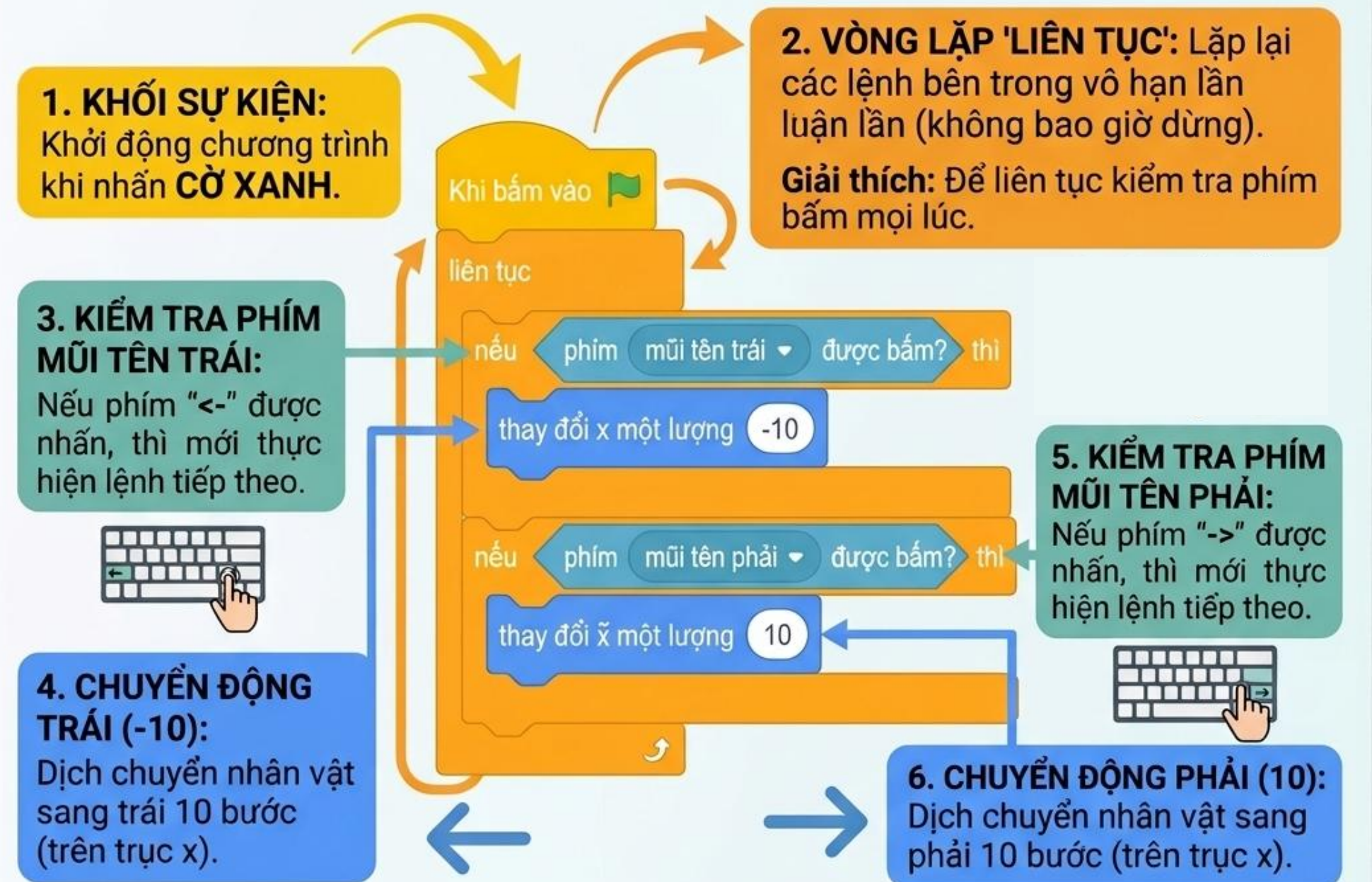
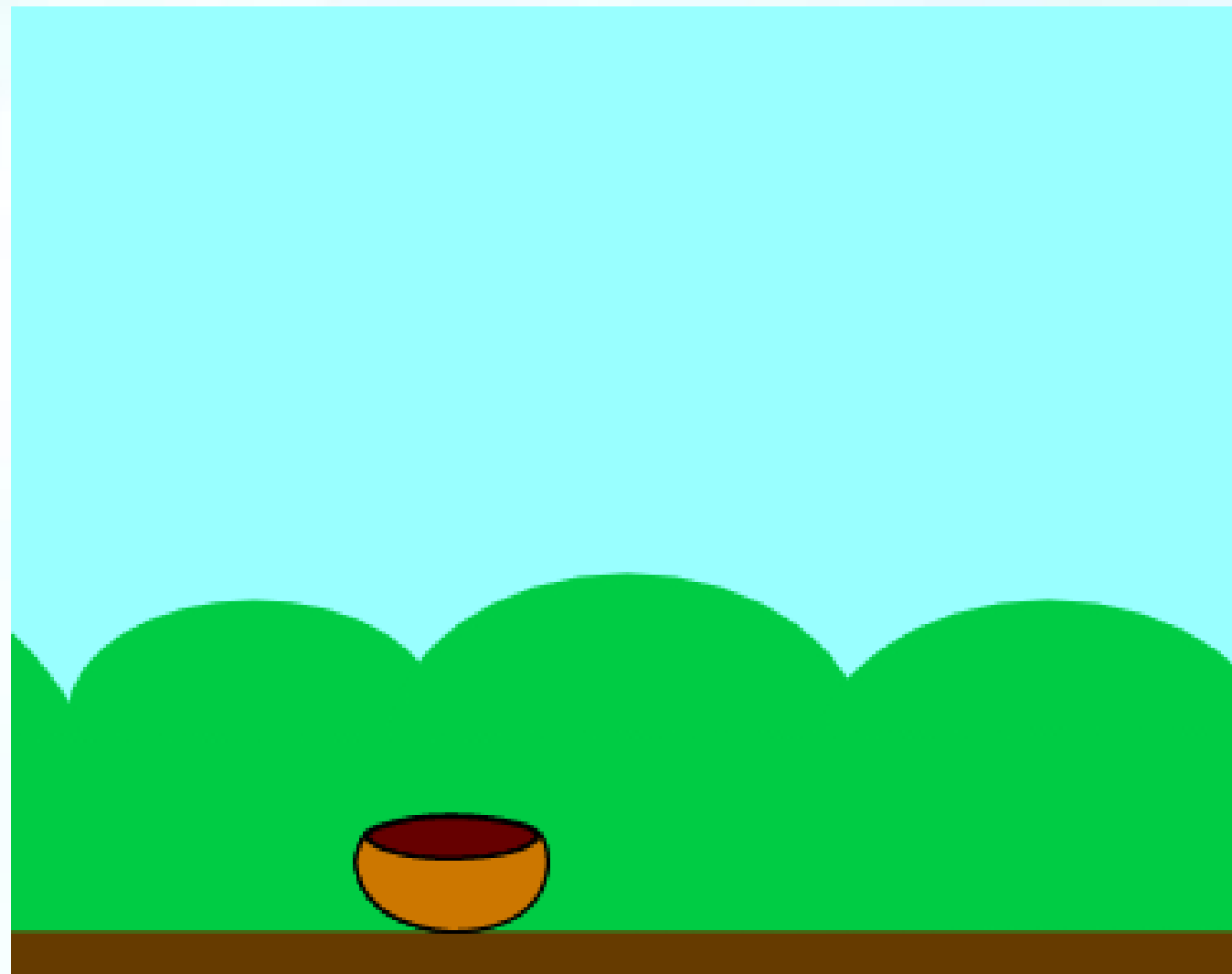
Mỗi bản sao là một loại quả khác nhau

Nếu chạm phải tô, quả bóng sẽ biến mất.

Nếu chạm vào cạnh dưới, game over

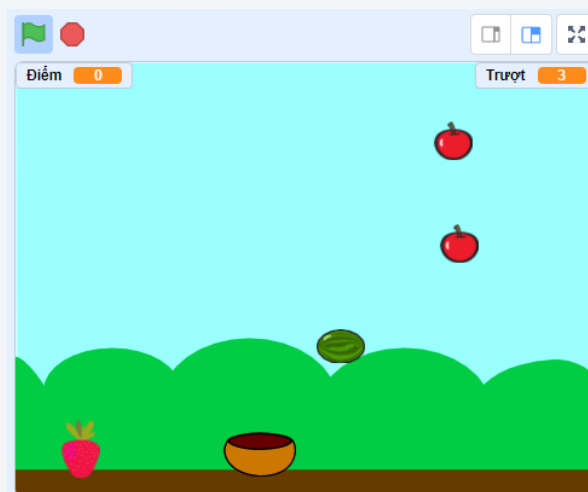
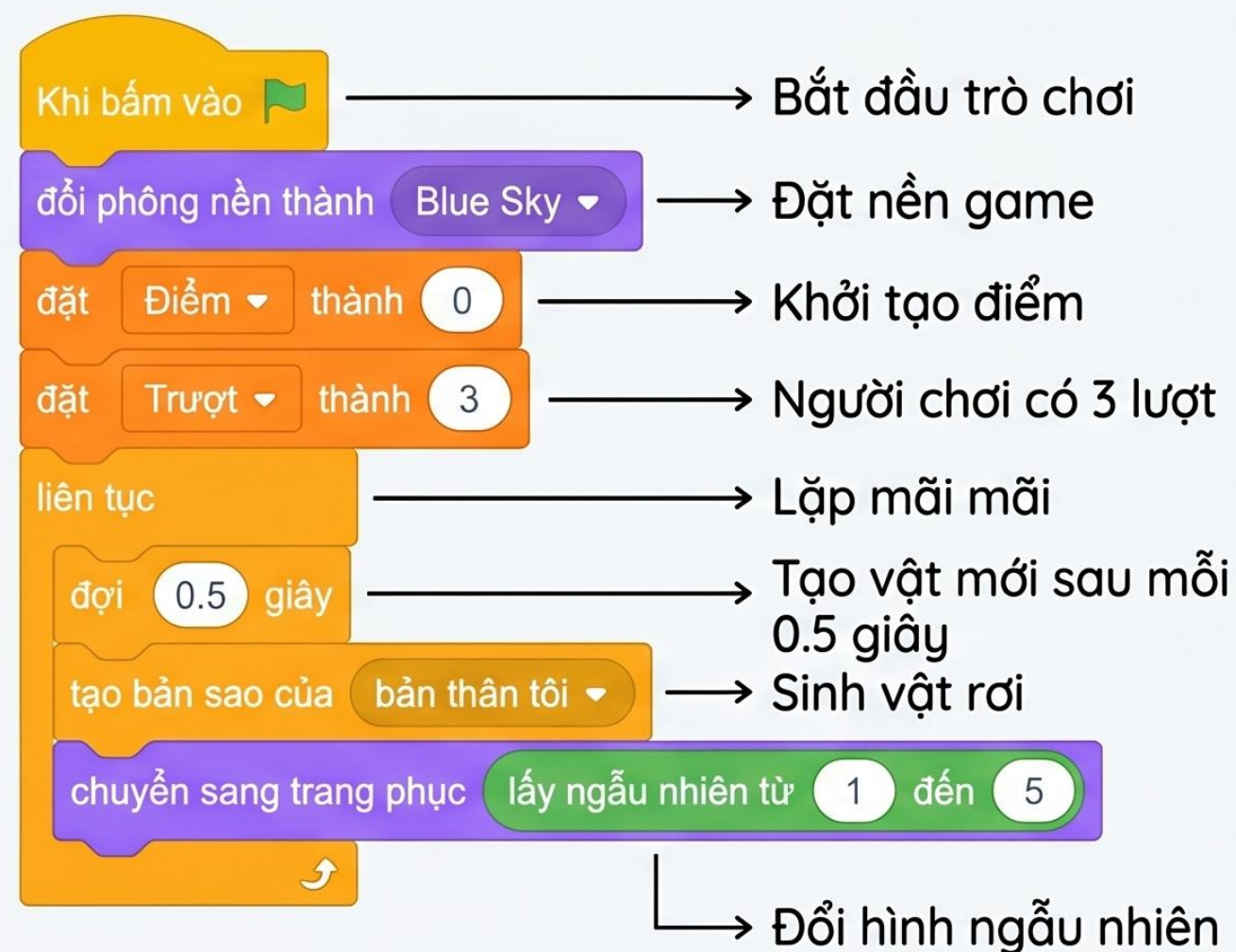


Bước 3.1 Lập trình điều khiển cái tô



Bước 3.2 Lập trình cho trái cây

1. Tạo vật rơi liên tục



2. Xử lý khi vật rơi chạm tô



Bài 12. Trò chơi đánh bóng bàn

Mục tiêu bài học

- Biết điều khiển hai thanh trượt để đỡ và đánh bóng.
- Hiểu cách làm bóng bật lại khi chạm cạnh sân hoặc thanh trượt.
- Biết tính điểm khi đối thủ không đỡ được bóng.
- Thực hiện giới hạn thời gian 60 giây và xác định người chiến thắng.
- Rèn luyện kỹ năng xử lý va chạm, tính điểm và tạo trò chơi hai người chơi trong Scratch.

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Trò chơi mô phỏng trận bóng bàn giữa hai người chơi. Mỗi người điều khiển một thanh trượt để đỡ và đánh bóng. Quả bóng sẽ bật lại khi chạm cạnh sân hoặc thanh trượt của đối thủ. Nếu một bên không đỡ được bóng, bên kia sẽ ghi điểm. Sau 60 giây, đội nào nhiều điểm hơn sẽ chiến thắng.



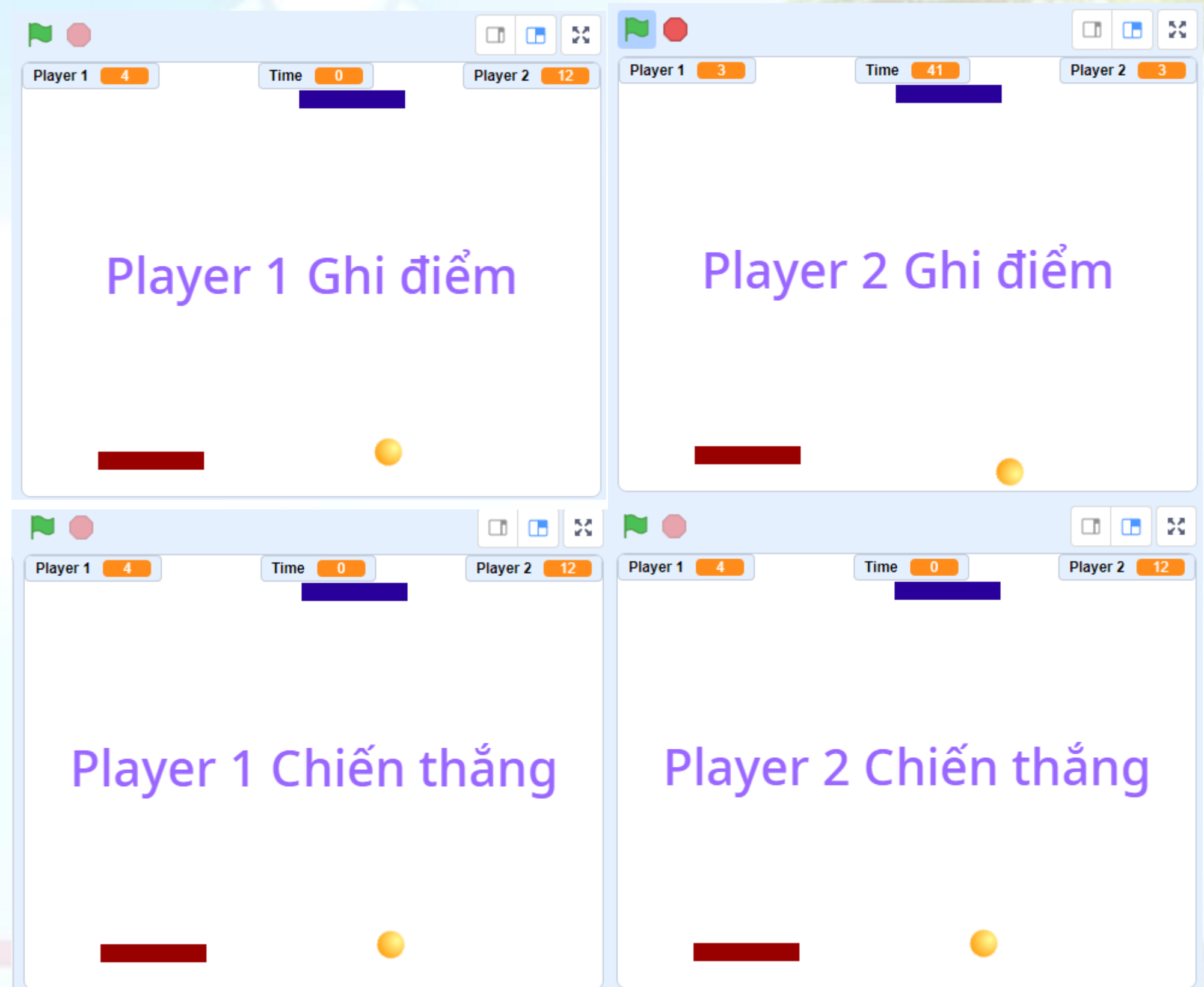
2) Các Bước Thực Hiện

Bước 1: Tạo phong nền:

- + Phong nền trò chơi
- + Phong nền lúc ghi điểm
- + Phong nền lúc chiến thắng

Bước 2: Nhân vật:

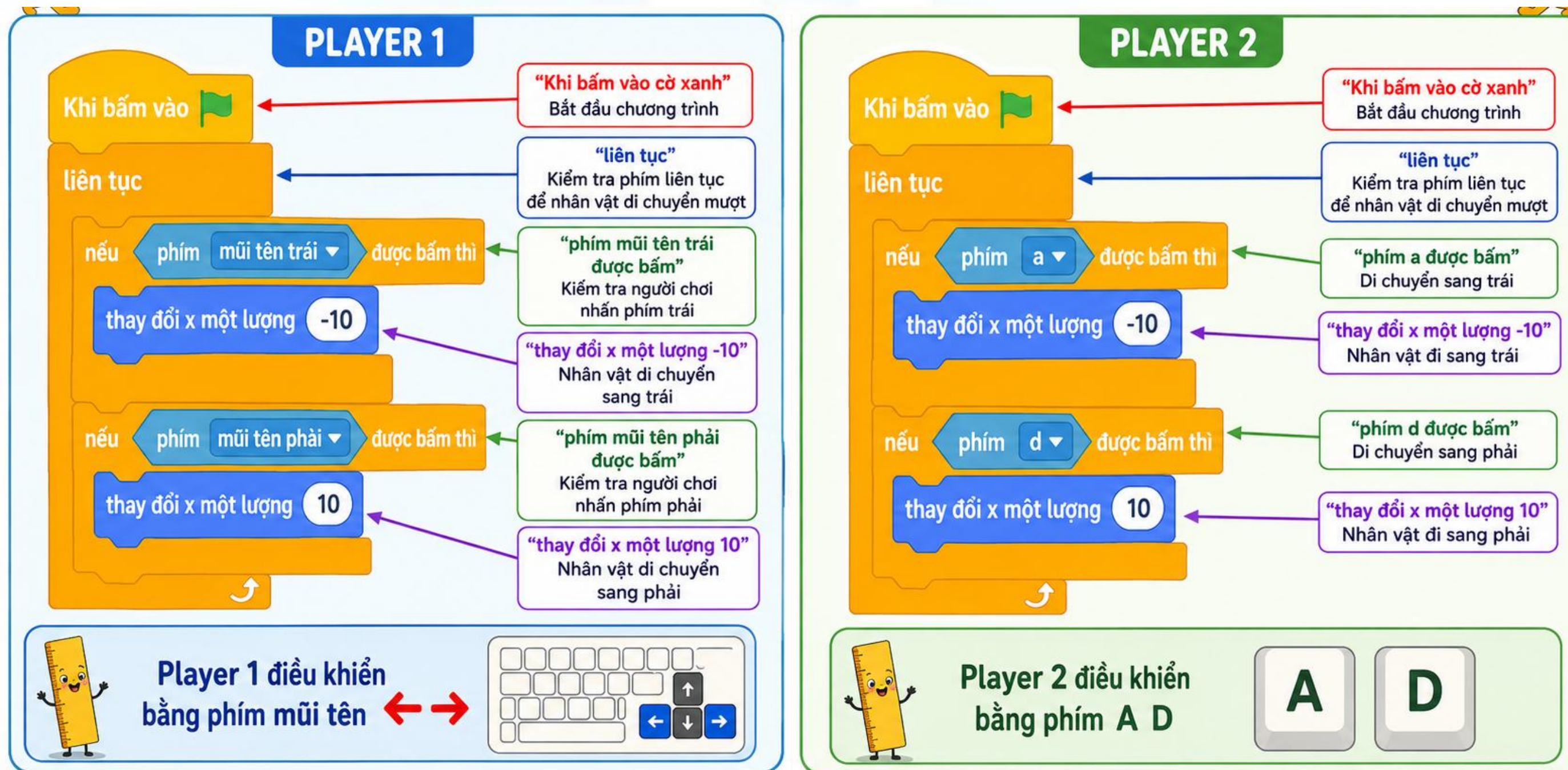
- + Player 1, 2
- + Quả bóng



Bước 3: Lập trình cho nhân vật

Bước 3.1 Lập trình cho 2 Player

- Player 1: Điều khiển trái phải bằng phím mũi tên Trái - Phải
- Player 2: Điều khiển trái phải bằng phím A - D

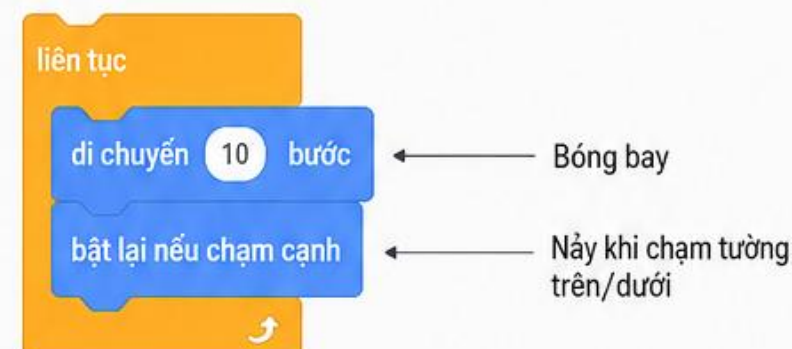


Bước 3.2 Lập trình cho quả bóng

1 CỤM 1: KHỞI TẠO TRÒ CHƠI



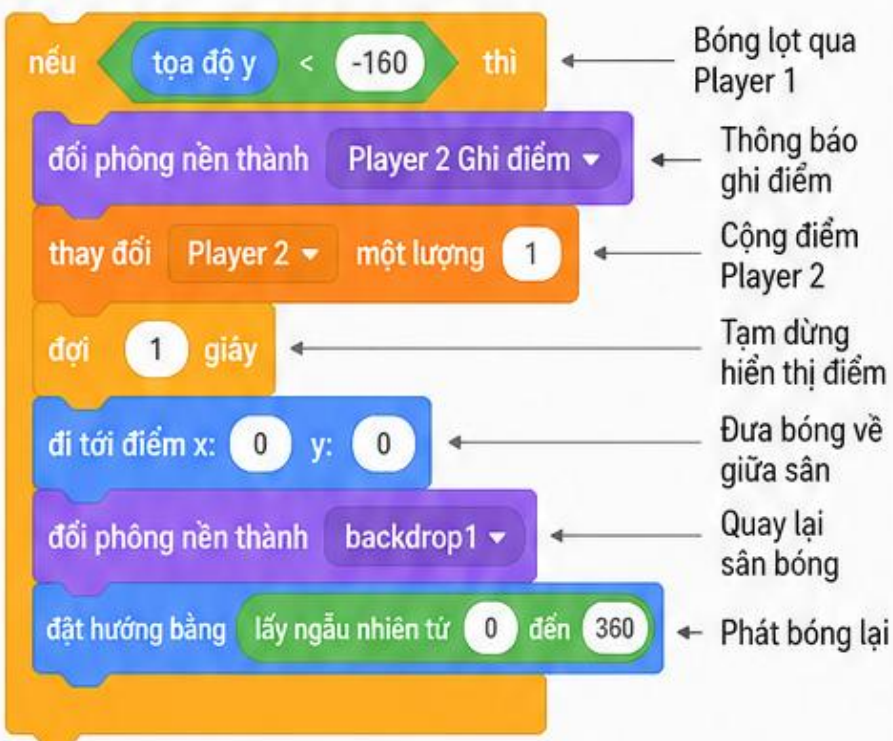
2 CỤM 2: BÓNG DI CHUYỂN LIÊN TỤC



3 CỤM 3: VA CHẠM THANH PLAYER 1



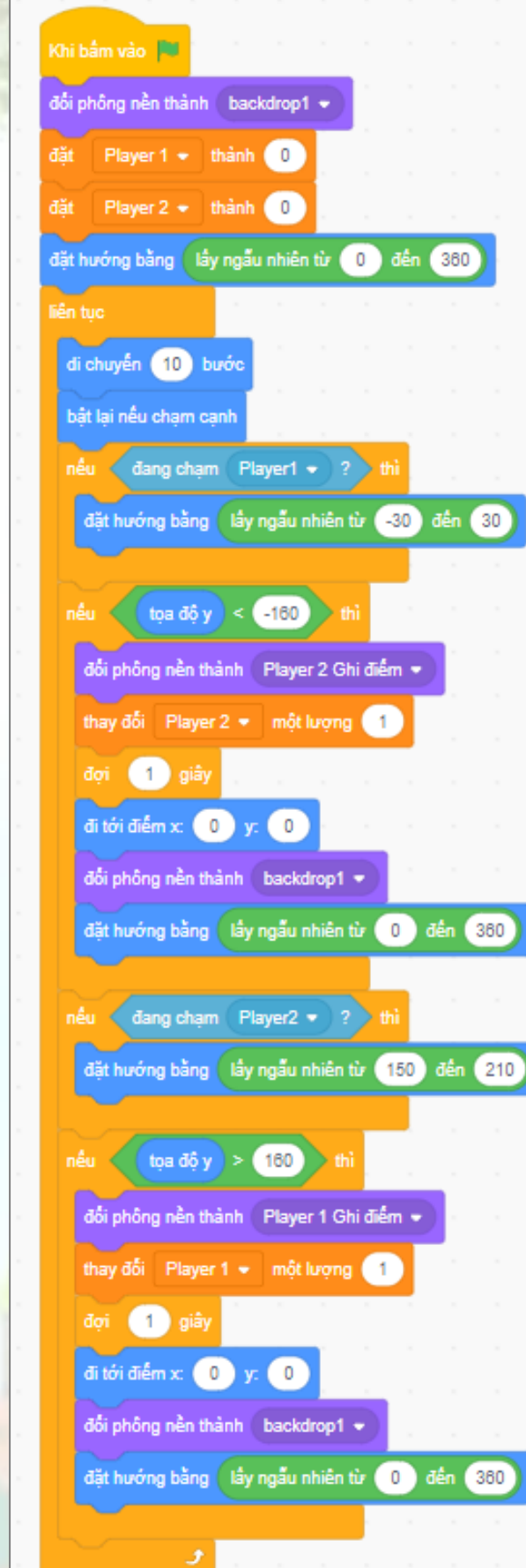
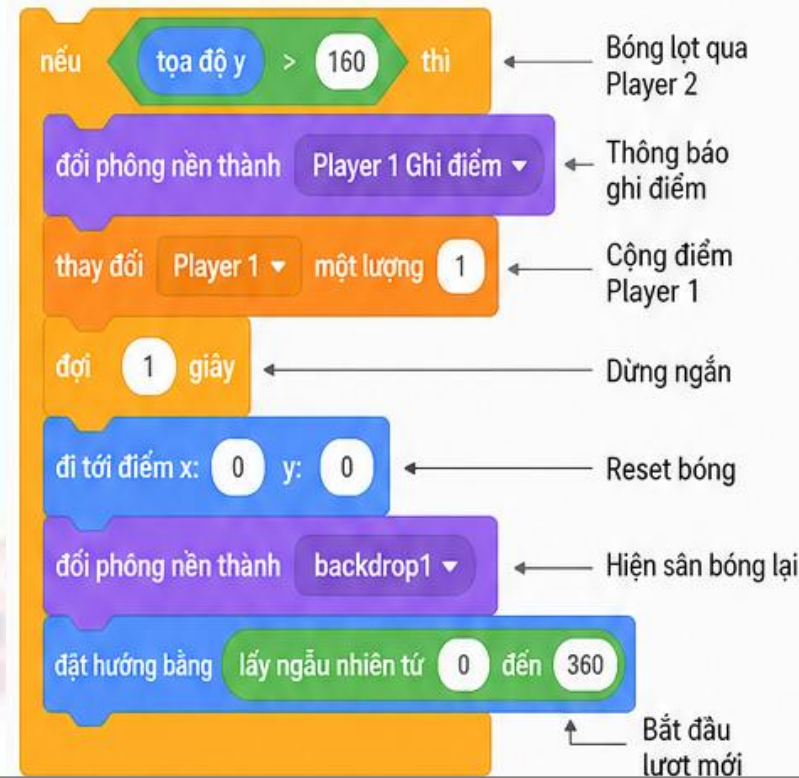
4 CỤM 4: PLAYER 2 GHI ĐIỂM



5 CỤM 5: VA CHẠM THANH PLAYER 2



6 CỤM 6: PLAYER 1 GHI ĐIỂM



Bước 3.3 Lập trình cho phong nền

Sau 60 giây, Xác định người chơi chiến thắng và dừng chương trình.



- 1 **Khối “Khi bấm vào cờ xanh”**
→ Bắt đầu trò chơi khi người chơi nhấn cờ xanh.
- 2 **Khối “đặt Time thành 60”**
→ Đặt thời gian bắt đầu của trận đấu là 60 giây.
- 3 **Khối “liên tục”**
→ Chương trình chạy lặp lại liên tục để cập nhật thời gian.
- 4 **Khối “đợi 1 giây”**
→ Mỗi lần chờ đúng 1 giây trước khi giảm thời gian.
- 5 **Khối “thay đổi Time một lượng -1”**
→ Giảm biến thời gian xuống 1 giây sau mỗi vòng lặp.
- 6 **Khối “nếu Time = 0 thì”**
→ Khi thời gian đếm ngược kết thúc, trò chơi sẽ kiểm tra người chiến thắng.
- 7 **Khối “nếu Player 1 < Player 2 thì”**
→ So sánh điểm số của hai người chơi.
- 8 **Khối “đổi phong nền thành Player 2 chiến thắng”**
→ Nếu Player 2 có điểm cao hơn, hiển thị màn hình chiến thắng của Player 2.
- 9 **Khối “đổi phong nền thành Player 1 chiến thắng”**
→ Nếu Player 1 có điểm cao hơn hoặc bằng, hiển thị màn hình chiến thắng của Player 1.
- 10 **Khối “dừng lại tất cả”**
→ Kết thúc toàn bộ trò chơi sau khi xác định người thắng.



Bài 13. Trò chơi bắn chim

Mục tiêu bài học

- Biết tạo và điều khiển xuất hiện ngẫu nhiên của các con chim trên màn hình.
- Hiểu cách phát hiện và xử lý khi bắn trúng mục tiêu (phát âm thanh, làm chim rơi xuống).
- Biết thêm hiệu ứng giạt súng và tia lửa khi bắn để trò chơi sinh động hơn.
- Thực hiện giới hạn thời gian 60 giây và tính điểm dựa trên số chim bắn hạ.
- Rèn luyện kỹ năng lập trình tương tác, phát hiện va chạm và xử lý hiệu ứng trong Scratch.

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Bạn sẽ vào vai một thợ săn và sẵn sàng bắn hạ tất cả con chim nào xuất hiện trên màn hình

- Chim sẽ xuất hiện ngẫu nhiên và di chuyển qua lại trên màn hình, Nếu bạn bắn trúng, phát ra âm thanh “quoác” và rơi xuống!
- Để tăng tính chân thực, Thêm hiệu giật súng và tia lửa nòng khi bắn.
- Bạn có 60s để bắn hạ tất cả lũ chim số điểm bạn đạt được chính là số chim bạn đã bắn hạ



2) Các Bước Thực Hiện

Bước 1: Phong nền trò chơi

- + Phong nền khi bắt đầu
- + Phong nền khi kết thúc trò chơi

Bước 2: Nhân vật

- + Tâm bắn
- + Các con chim
- + Khẩu súng

Bước 3: Âm thanh

- + Nhạc nền
- + Tiếng súng
- + Tiếng chim khi bị bắn trúng
- + Âm thanh kết thúc trò chơi



Bước 4: Lập trình cho nhân vật

Bước 4.1 Tâm bắn:

- Liên tục di chuyển đến vị trí trỏ chuột



LẬP TRÌNH TÂM BẮN THEO CHUỘT TRONG SCRATCH



Tâm bắn luôn theo vị trí chuột và tạo hiệu ứng rung/giật khi người chơi bắn.

1. ĐIỀU KHIỂN VỊ TRÍ TÂM BẮN



GIẢI THÍCH:
Tâm bắn luôn bám theo chuột nhờ khối “đi tới con trỏ chuột” được lặp lại liên tục.

2. HIỆU ỨNG BẮN / GIẬT CỦA TÂM BẮN



GIẢI THÍCH:
Khi bắn (nhấn chuột), tâm bắn đổi nhanh trang phục để tạo cảm giác giật. Khi thả chuột, tâm bắn trở lại trạng thái bình thường.

 **MẸO NHỎ:** Bạn có thể tạo nhiều trang phục cho tâm bắn (ví dụ: to – nhỏ – sáng) để hiệu ứng giật trông sinh động hơn!



Bước 4.2: Các con chim:

- Các con chim liên tục tạo bản sao mỗi 5s
- Khi bắt đầu bản sao, Liên tục di chuyển theo chiều ngang trên màn hình
- Nếu bị bắn trúng, hướng 180 độ và rơi xuống
- Biến mất nếu chạm cạnh



Bài 14. Trò chơi đoán số

Mục tiêu bài học

- Biết tạo số ngẫu nhiên trong khoảng từ 1 đến 10.
- Hiểu cách nhập và so sánh câu trả lời của người chơi với số mà nhân vật nghĩ ra.
- Biết hiển thị phản hồi phù hợp khi người chơi đoán đúng hoặc sai.
- Rèn luyện kỹ năng làm việc với dữ liệu nhập vào, điều kiện rẽ nhánh và phản hồi nhân vật trong Scratch.

Nội dung nghiên cứu

- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

- Nhân vật chú mèo sẽ nghĩ ra ngẫu nhiên 1 con số từ 1 đến 10 và hỏi: “đoán xem mình đang nghĩ đến số mấy nào?”
- Bạn sẽ phải đoán xem chú mèo đang nghĩ đến số mấy và trả lời.
- Nếu bạn trả lời đúng chú mèo sẽ nói: “chính xác!”, ngược lại nói “sai rồi! Mình đang nghĩ đến số [số mà chú mèo nghĩ]”



2) Các Bước Thực Hiện

Bước 1: Phong nền trò chơi

Bước 2: Nhân vật : Chú mèo

Bước 3: Lập trình cho nhân vật chú mèo



Bài 15. Trò chơi đua xe

Mục tiêu bài học

- Biết điều khiển xe bằng các phím mũi tên Trái – Phải.
- Hiểu cách tạo chuyển động liên tục cho xe và các phương tiện khác.
- Biết phát hiện va chạm giữa các xe và kết thúc trò chơi khi xảy ra va chạm.
- Rèn luyện kỹ năng lập trình điều khiển, phát hiện va chạm và xử lý tình huống trong trò chơi bằng Scratch.

Nội dung nghiên cứu

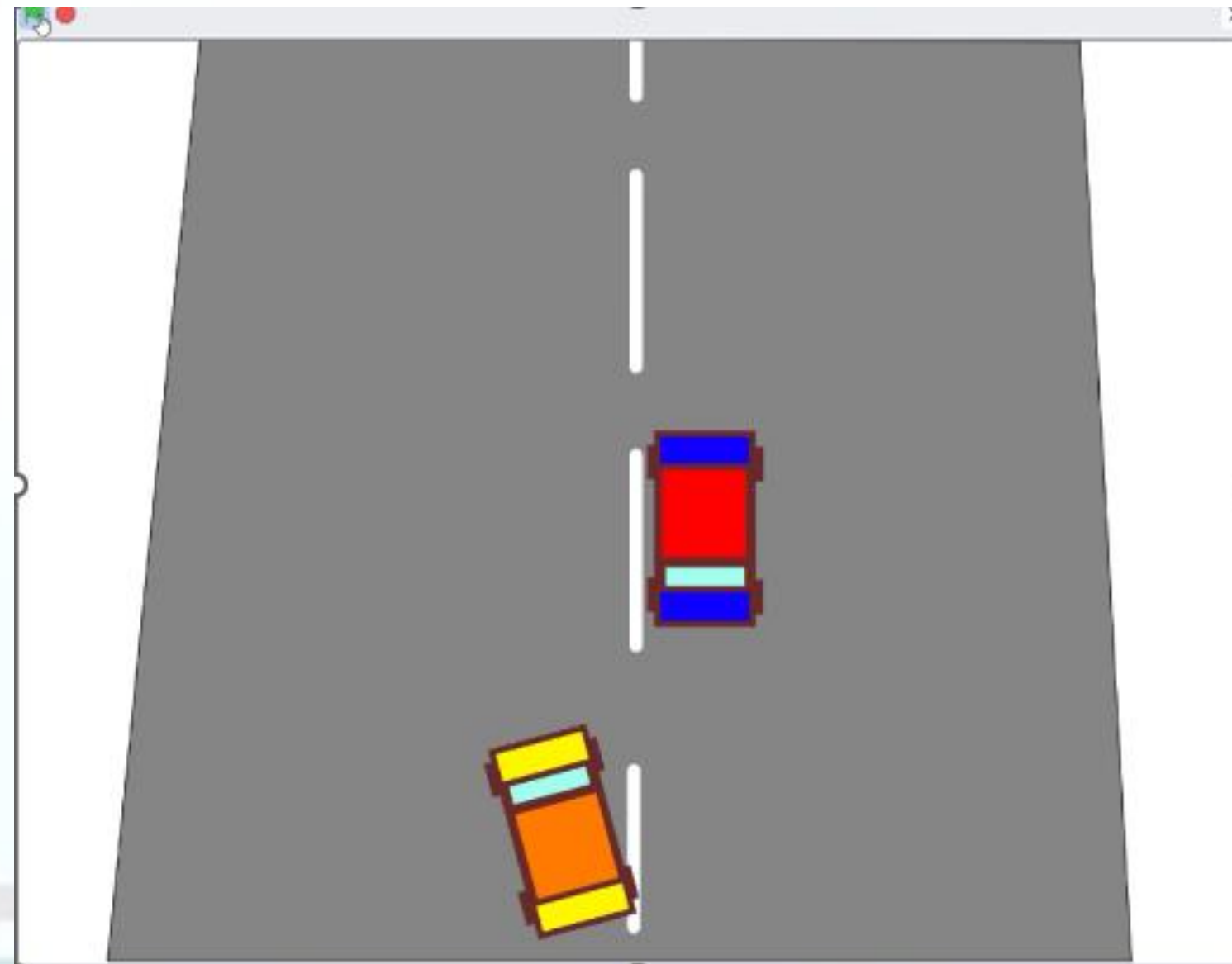
- 1) Tổng quan chương trình
- 2) Các bước thực hiện



1) Tổng quan chương trình

Bạn sẽ điều khiển chiếc xe bằng phím mũi tên Trái – Phải và tránh các phương tiện khác.

Nếu va chạm phải các phương tiện khác, trò chơi kết thúc



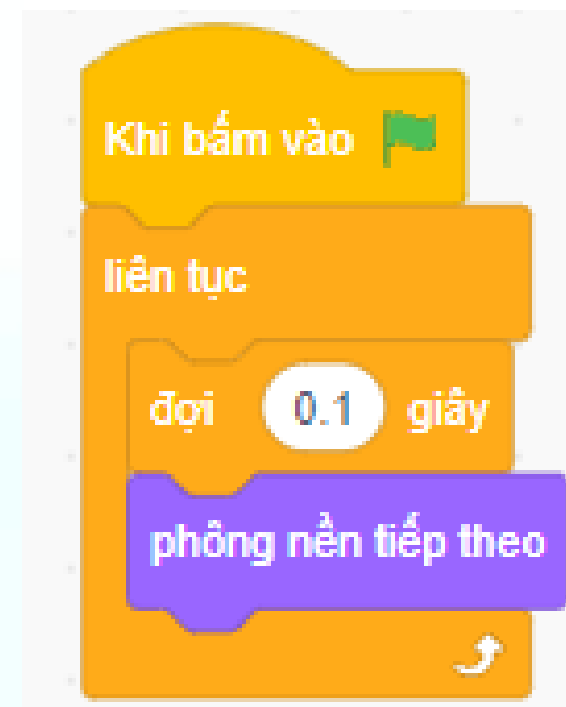
2) Các Bước Thực Hiện

Bước 1: Phong nền con đường, gồm nhiều trang phục để tạo hiệu ứng đang di chuyển

Bước 2: Nhân vật các chiếc xe

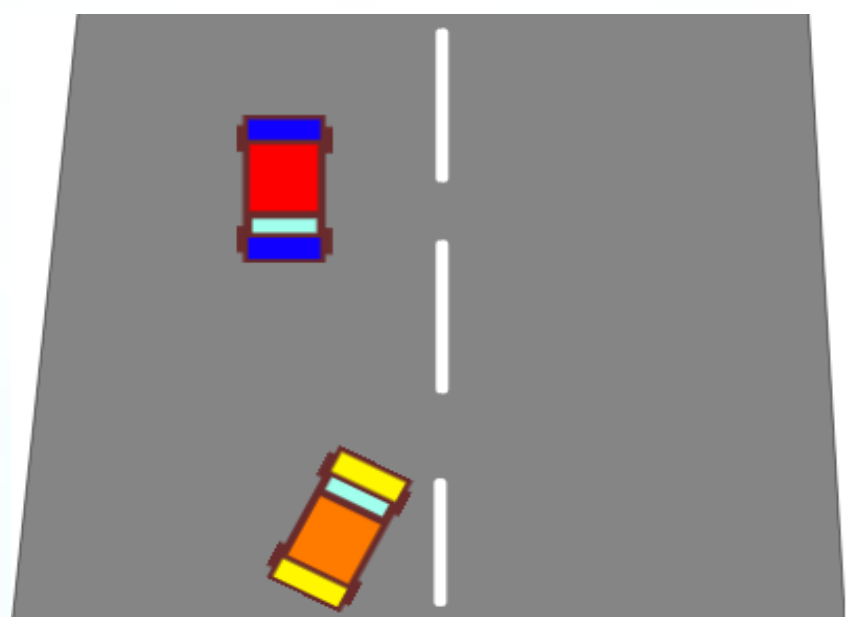
Bước 3: Lập trình

Bước 3.1. Phong nền: Liên tục chuyển đổi tạo hiệu ứng đang di chuyển

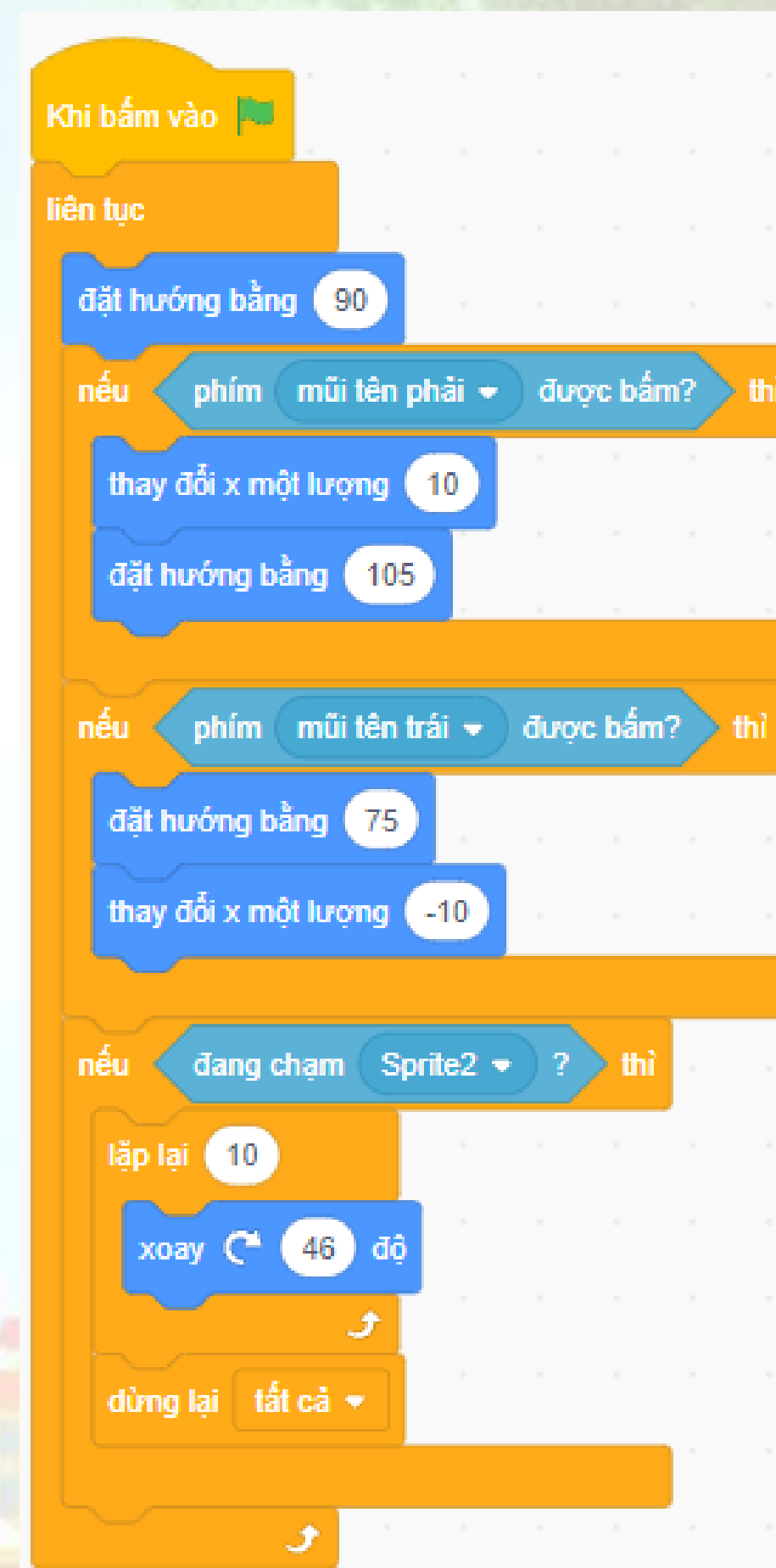


Bước 3.2. Chiếc xe của bạn:

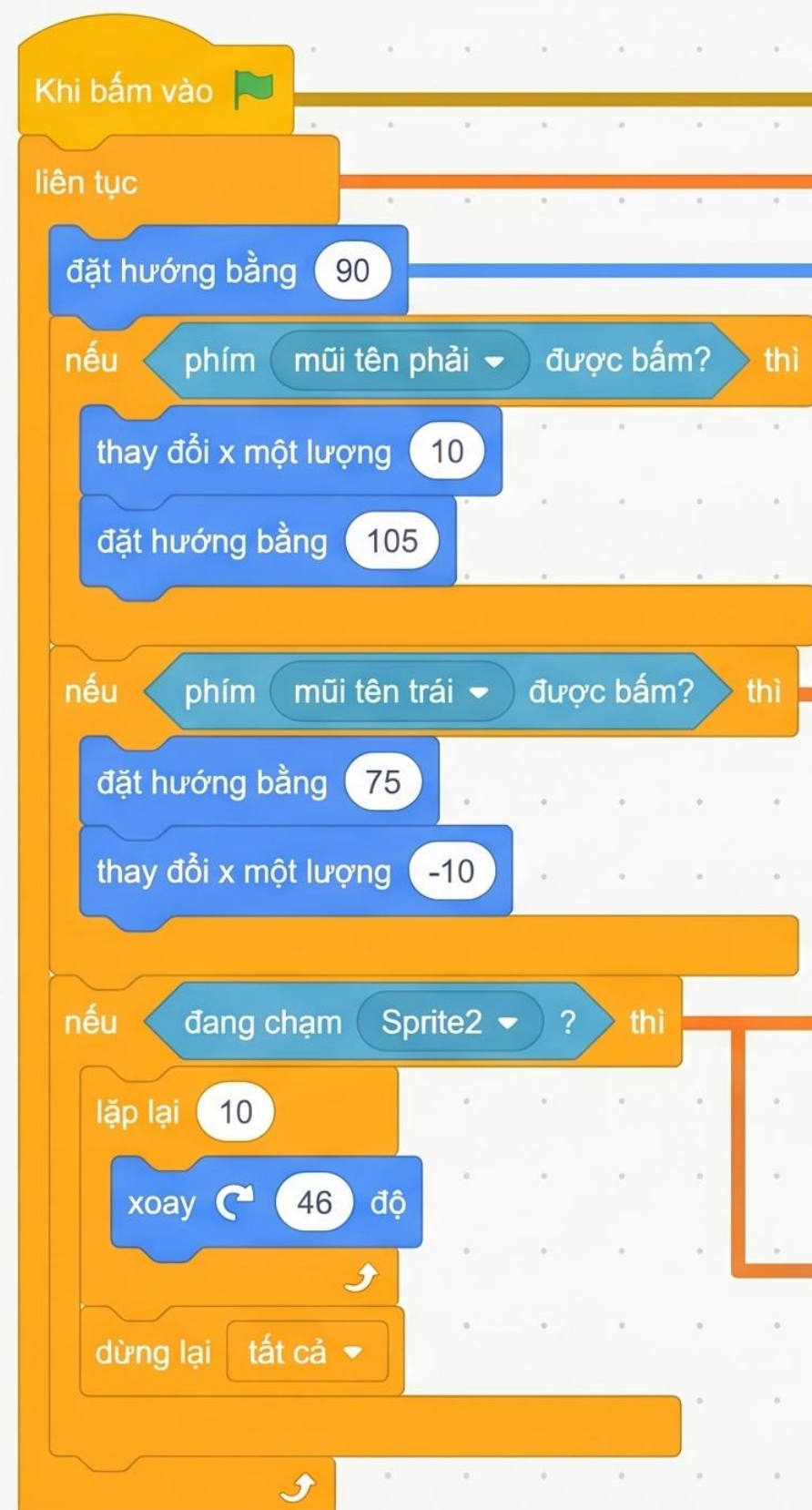
- Điều khiển sang trái, Phải bằng phím mũi tên.
- Để thêm hiệu ứng chân thực, chiếc xe sẽ nghiêng sang phải khi di chuyển sang phải, và nghiêng sang trái khi di chuyển sang trái



- Nếu va chạm phải phương tiện khác, xoay vài vòng và dừng chương trình



HƯỚNG DẪN MÃ ĐIỀU KHIỂN XE Ô TÔ (SCRATCH)



1. Bắt đầu khi nhấn cờ xanh

2. Vòng lặp chính, kiểm tra liên tục các hành động của xe

3. Đặt lại xe thẳng đứng (hướng 90 độ) mỗi khi bắt đầu một chu kỳ kiểm tra mới

4. Điều khiển **SANG PHẢI**: Thay đổi tọa độ x tăng lên 10 và đặt hướng thành 105 độ để tạo hiệu ứng xe nghiêng nhẹ sang phải khi rẽ

5. Điều khiển **SANG TRÁI**: Đặt hướng thành 75 độ để xe nghiêng nhẹ sang trái và giảm tọa độ x đi 10

6. Kiểm tra va chạm: Phát hiện nếu xe đụng phải phương tiện khác (Sprite2)

7. Xử lý va chạm: Xe sẽ xoay vòng vòng (10 lần x 46 độ = 460 độ) để tạo hiệu ứng lộn nhào, sau đó kết thúc toàn bộ trò chơi



Bước 3.3. Các phương tiện khác:

- Liên tục tạo các bản sao

- Các bản sao di chuyển từ trên cùng của sân khấu xuống dưới cùng của sân khấu, sẵn sàng va chạm với bạn

Khi bấm vào

1. Bắt đầu dự án.

liên tục

2. Vòng lặp vô hạn.

tạo bản sao của

bản thân tôi

3. Nhân bản Sprite chính.

đợi 0.5 giây

4. Tạo khoảng nghỉ giữa các bản sao.

***Tóm tắt*:** Sprite chính liên tục tạo ra một bản sao mới sau mỗi 0.5 giây, nhưng bản thân nó không di chuyển.

khi tôi bắt đầu là một bản sao

1. Bắt đầu hành động cho mỗi bản sao mới.

hiện

2. Hiển thị bản sao.

đi tới điểm x:

lấy ngẫu nhiên từ -145 đến 145

y: 140

Luôn bắt đầu ở trên cùng.

Vị trí ngang ngẫu nhiên.

liên tục

4. Chuyển động và kiểm tra của bản sao.

thay đổi y một lượng -10

5. Làm cho bản sao rơi xuống.

nếu

đang chạm

cạnh

?

6. Kiểm tra xem đã chạm đáy màn hình chưa.

ẩn

7. Làm cho bản sao biến mất.

***Tóm tắt*:** Mỗi bản sao mới xuất hiện ở một vị trí ngẫu nhiên trên đỉnh màn hình, rơi xuống cho đến khi chạm đáy và biến mất.

