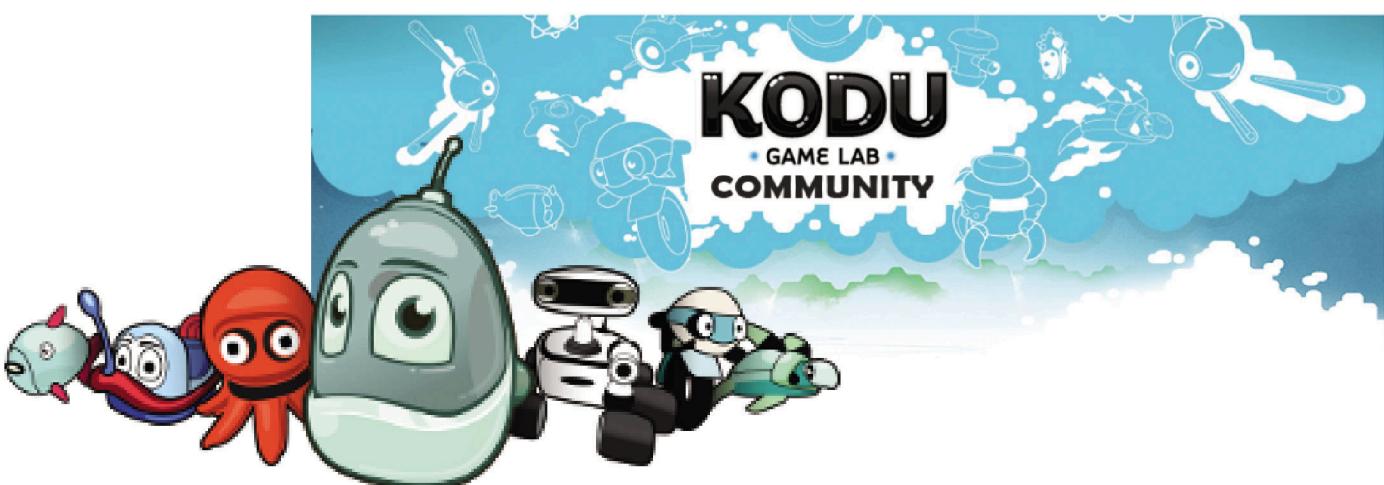


**CHƯƠNG TRÌNH TIN HỌC ỨNG DỤNG
VÀ KHOA HỌC MÁY TÍNH**



VIETNET-ICT

CHỦ ĐỀ: LẬP TRÌNH 2D VÀ 3D



**DỰ ÁN TĂNG CƯỜNG KỸ NĂNG CÔNG NGHỆ THÔNG TIN
CHO GIỚI TRẺ HỘI NHẬP VÀ PHÁT TRIỂN**



VIETNET-ICT

YOUTHSPARK DIGITAL INCLUSION

ĐÀO THÁI LAI (Chủ biên)
ĐỖ TRUNG TUẤN - NGUYỄN VĂN NAM - TÔ THỦY DIỄM QUYÊN

CHƯƠNG TRÌNH TIN HỌC ỨNG DỤNG VÀ
KHOA HỌC MÁY TÍNH

CHỦ ĐỀ: LẬP TRÌNH 2D VÀ 3D

(Tái bản lần thứ hai)

Đơn vị tài trợ

Microsoft Việt Nam

Đơn vị chủ trì

Cục Công nghệ thông tin -
Bộ Giáo dục và Đào Tạo

Đơn vị điều phối

Trung tâm Công nghệ thông tin -
Truyền thông Vietnet

Bản quyền thuộc Dự án Tăng cường kỹ năng công nghệ thông tin
cho giới trẻ hội nhập và phát triển

MỤC LỤC

PHẦN 1. LẬP TRÌNH 2D VỚI SCRATCH.....	7
LỜI NÓI ĐẦU.....	5
Mô đun 1. BẮT ĐẦU VỚI SCRATCH.....	9
1. Dự án.....	9
2. Giao diện.....	10
3. Lập trình.....	12
4. Khối lệnh (Block).....	14
5. Biên tập âm thanh và hình vẽ.....	16
6. Khung đối tượng.....	17
7. Tạo mới một đối tượng.....	18
8. Phần mở rộng – Thực hành.....	18
Mô đun 2. BÀI TOÁN CHUYỂN ĐỘNG.....	20
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	20
1. Trả lời các câu hỏi.....	20
2. Tìm hiểu các nhóm lệnh.....	21
2.1. Nhóm lệnh chuyển động.....	21
2.2. Nhóm lệnh trượt, thay đổi vị trí và định vị.....	24
2.3. Thay đổi trang phục và nền sân khấu.....	29
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	31
C. HOẠT ĐỘNG SÁNG TẠO.....	32
Mô đun 3. HỘI THOẠI VÀ HIỆU ỨNG ĐỒ HOẠ ĐỐI TƯỢNG.....	33
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	33
1. Mở bài tập “Di Chuyen 4”.....	33
2. Trả lời câu hỏi.....	34
3. Tìm hiểu các nhóm lệnh.....	34
3.1. Đối tượng suy nghĩ và nói.....	35
3.2. Hiệu ứng đồ họa.....	37
3.3. Thay đổi kích thước đối tượng, thứ tự các lớp.....	40
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	34
C. HOẠT ĐỘNG SÁNG TẠO.....	46
Mô đun 4. CHƠI NHẠC VÀ VẼ HÌNH TRONG SCRATCH.....	48
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	48

1. Quan sát và trả lời câu hỏi.....	48
2. Tìm hiểu các nhóm lệnh.....	49
2.1.Chơi nhạc.....	49
2.2.Chơi nhạc cụ.....	51
2.3.Vẽ hình và kiểm soát màu.....	52
2.4.Thay đổi kích thước, sắc thái nét vẽ và hiệu ứng in hình.....	56
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	58
C. HOẠT ĐỘNG SÁNG TẠO.....	59
 Mô đun 5. ĐIỀU KHIỂN VÀ CẤU TRÚC ĐIỀU KHIỂN.....	 61
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	61
1. Thủ nghiệm các lệnh.....	61
2. Tìm hiểu các nhóm lệnh.....	61
2.1.Nhóm lệnh sự kiện.....	61
2.2.Nhóm cấu trúc điều khiển.....	64
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	71
C. HOẠT ĐỘNG SÁNG TẠO.....	72
 Mô đun 6. TOÁN HỌC TRONG SCRATCH.....	 74
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	74
1. Quan sát.....	74
2. Tính điểm.....	75
3. Tìm hiểu các nhóm lệnh.....	75
3.1.Nhóm lệnh toán học.....	75
3.2.Sử dụng biến trong Scratch.....	77
3.3.Sử dụng danh sách trong Scratch.....	80
3.4.Nhóm lệnh cảm biến.....	82
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	85
C. HOẠT ĐỘNG SÁNG TẠO.....	87
 PHẦN 2. LẬP TRÌNH 3D VỚI KODU GAME LAB.....	 89
Mô đun 1. SỬ DỤNG KODU ĐỂ TẠO CẢNH HOẠT HÌNH.....	91
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	91
1. Tìm hiểu đặc điểm của trò chơi điện tử.....	91
2. Chạy và tìm hiểu một số chức năng của chương trình KODU.....	92
3. Xây dựng địa hình.....	93
4. Lưu chương trình.....	94
5. Tạo đối tượng, nhân vật cho trò chơi.....	95
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	97
C. HOẠT ĐỘNG SÁNG TẠO.....	99

Mô đun 2. THIẾT KẾ TRÒ CHƠI VỚI PHẦN NỀN PHỨC TẠP CÓ 2, 3 ĐỐI TƯỢNG/ NHÂN VẬT.....	100
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	100
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	103
1. Thay đổi lệnh.....	103
2. Cho KODU chạy theo đường dẫn.....	104
3. Thêm đối tượng và hành vi	104
4. Thực hiện theo kịch bản có sẵn.....	104
C. HOẠT ĐỘNG SÁNG TẠO.....	105
1. Lập trình hành vi của KODU.....	105
2. Tạo kịch bản đơn giản.....	105
3. Viết chương trình.....	105
Mô đun 3. THIẾT KẾ TRÒ CHƠI CÓ ĐIỀU KHIỂN, LẬP TRÌNH HÀNH VI CỦA ĐỐI TƯỢNG CÓ TƯƠNG TÁC.....	106
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	106
1. Tìm hiểu, phân tích trò chơi có sẵn.....	106
2. Tìm hiểu lệnh chọn hành vi cho KODU.....	107
3. Tìm hiểu về các điều kiện có thể xảy ra cho KODU.....	107
4. Điều khiển KODU bằng bàn phím.....	108
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	109
C. HOẠT ĐỘNG SÁNG TẠO.....	111
1. Sáng tạo trò chơi mới.....	111
2. Trình diễn sản phẩm.....	111
Mô đun 4. TÌM HIỂU CÁC LỆNH LẬP TRÌNH VÀ GIỚI THIỆU SẢN PHẨM VỚI CỘNG ĐỒNG.....	112
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	112
1. Chèn âm thanh, nhạc vào không gian hoạt cảnh/ trò chơi.....	112
2. Tìm hiểu các trang.....	112
3. Tạo trang mới.....	113
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	114
1. Luyện tập các lệnh.....	114
2. Phân tích các yếu tố của trò chơi.....	114
C. HOẠT ĐỘNG SÁNG TẠO.....	116
Mô đun 5. LUYỆN TẬP NÂNG CAO.....	118
A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ.....	118
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG.....	121
C. HOẠT ĐỘNG SÁNG TẠO.....	116

PHẦN 1.

LẬP TRÌNH 2D VỚI SCRATCH

LỜI NÓI ĐẦU

Nội dung “**Lập trình 2D với Scratch**” được biên soạn với mục đích giúp học sinh trung học cơ sở bước đầu làm quen với các khái niệm trong lập trình. Phát triển theo hướng tương tác trực quan với các thiết kế đồ họa sống động và dễ vận dụng vào các tình huống cụ thể, lập trình Scratch được xem là ngôn ngữ nền tảng trước khi học các ngôn ngữ lập trình khác.

Khi sử dụng ngôn ngữ lập trình Scratch, thay vì phải viết những dòng lệnh logic dễ gặp lỗi, bạn chỉ cần kéo và thả các khối lệnh đầy màu sắc có sẵn để lắp ghép thành một kịch bản điều khiển các đối tượng trên sân khấu biểu diễn.

Tài liệu này được biên soạn theo 6 mô đun:

Mô đun 1 - Bắt đầu với Scratch

Mô đun 2 - Bài toán chuyển động

Mô đun 3 - Hội thoại và hiệu ứng đồ họa nhân vật

Mô đun 4 - Chơi nhạc và vẽ hình

Mô đun 5 - Điều khiển và cấu trúc điều khiển

Mô đun 6 - Toán học trong Scratch

Mỗi mô đun có cấu trúc như sau:

A. *Hoạt động tìm tòi, khám phá*: Học sinh chủ động khám phá các kiến thức mới trong môn học, thông qua việc thử nghiệm các câu lệnh, chức năng trong Scratch, từ đó đưa ra những nhận xét.

B. *Hoạt động thực hành, ứng dụng*: Học sinh sẽ được luyện tập, củng cố những kiến thức, kỹ năng, phương pháp đã khám phá được ở hoạt động A. Có thêm một số tình huống yêu cầu phải ứng dụng trong điều kiện tương đối mới.

C. *Hoạt động sáng tạo*: Học sinh vận dụng các kiến thức, kỹ năng đã phát hiện được để thực hiện một dự án học tập. Dự án sẽ nêu chủ đề các yêu cầu cần về kịch bản trò chơi, cách thức chơi. Học sinh có thể làm việc theo cặp, nhóm, sau đó trình bày, bảo vệ sản phẩm trước giáo viên và các bạn.

Nội dung này đồng thời được số hóa tại địa chỉ <http://digitalskill.vn>. Dưới sự hướng dẫn của các giáo viên, học sinh có thể tự học theo các tài liệu để sáng tạo được nhiều hoạt hình thú vị. Việc đánh giá kết quả học tập được thực hiện qua đánh giá thường xuyên và đánh giá kết quả thực hiện dự án.

Hy vọng rằng “Lập trình 2D với Scratch” sẽ giúp bạn học Tin học một cách chủ động và sáng tạo, phát triển kỹ năng và tư duy một cách toàn diện.

Mô đun 1. **BẮT ĐẦU VỚI SCRATCH**

Mặc dù Scratch được thiết kế cho những người hoàn toàn mới bắt đầu làm quen với lập trình, đây vẫn có thể là một chương trình lập trình đầy thử thách và có tính ứng dụng cao. Cấu trúc của Scratch có tính phức tạp riêng, bao gồm các kịch bản, vẽ, trình soạn thảo, chỉnh sửa âm thanh,... Bài học này cung cấp các khái niệm đầu tiên về Scratch để ngay cả những bạn hoàn toàn mới và chưa thực sự tiếp cận với bất cứ một ngôn ngữ lập trình nào cũng có thể làm quen.

Bạn có thể truy cập <https://scratch.mit.edu/download> để tải xuống phần mềm, cài đặt và bắt đầu tìm hiểu sử dụng. Các nội dung trong tài liệu này hướng dẫn các thao tác đối của phiên bản Scratch 2.0 v461.

Đồng thời, bạn có thể dùng Scratch trực tuyến tại: <https://scratch.mit.edu/projects/editor>

MỤC TIÊU

- Tìm hiểu một số khái niệm về Scratch.
- Làm quen với giao diện và các thành phần trên màn hình.
- Làm quen với môi trường lập trình kéo thả.
- Thêm, sửa, xoá đối tượng và sân khấu.
- Tìm hiểu các lệnh Scratch.
- Thực hiện lưu một dự án.

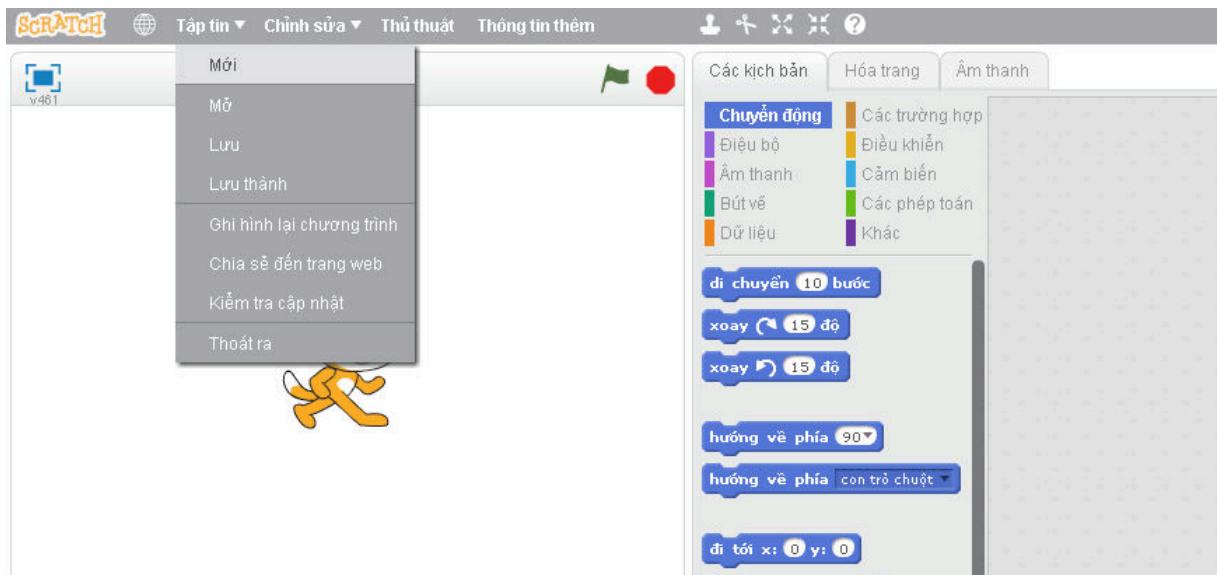
1. Dự án

Các dự án về *hoạt hình, truyện, trò chơi* - và tất cả mọi thứ đều có thể được thực hiện tại Scratch! Các dự án được thực hiện bằng chương trình Scratch có thể được chia sẻ với cộng đồng Scratch trên toàn thế giới thông qua Internet.



Các sản phẩm là ứng dụng được tạo ra bởi Scratch có thể hoạt động tốt trên máy tính hoặc được chia sẻ trên Internet thông qua các tài khoản được đăng ký trước.

- Để tạo mới 1 dự án bạn chọn **Tập tin**, sau đó chọn **Mới**
- Để lưu 1 dự án bạn chọn **Tập tin**, sau đó chọn **Lưu** (đối với bản sử dụng trực tuyến bạn vào **Tập tin**, sau đó chọn **Tải xuống máy tính của bạn**)

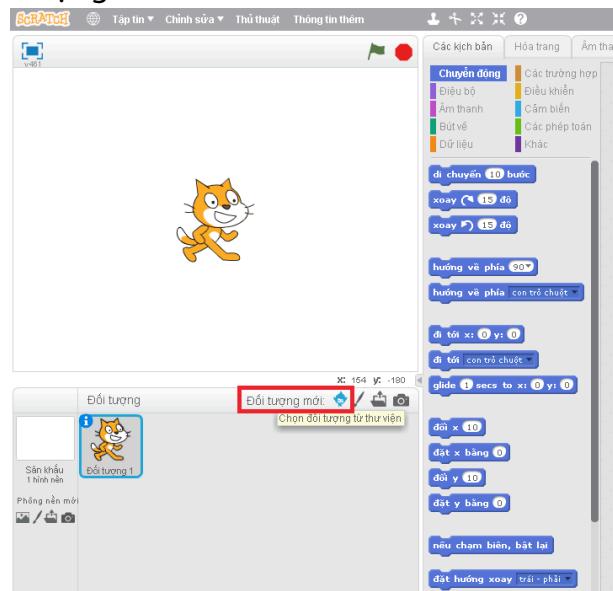


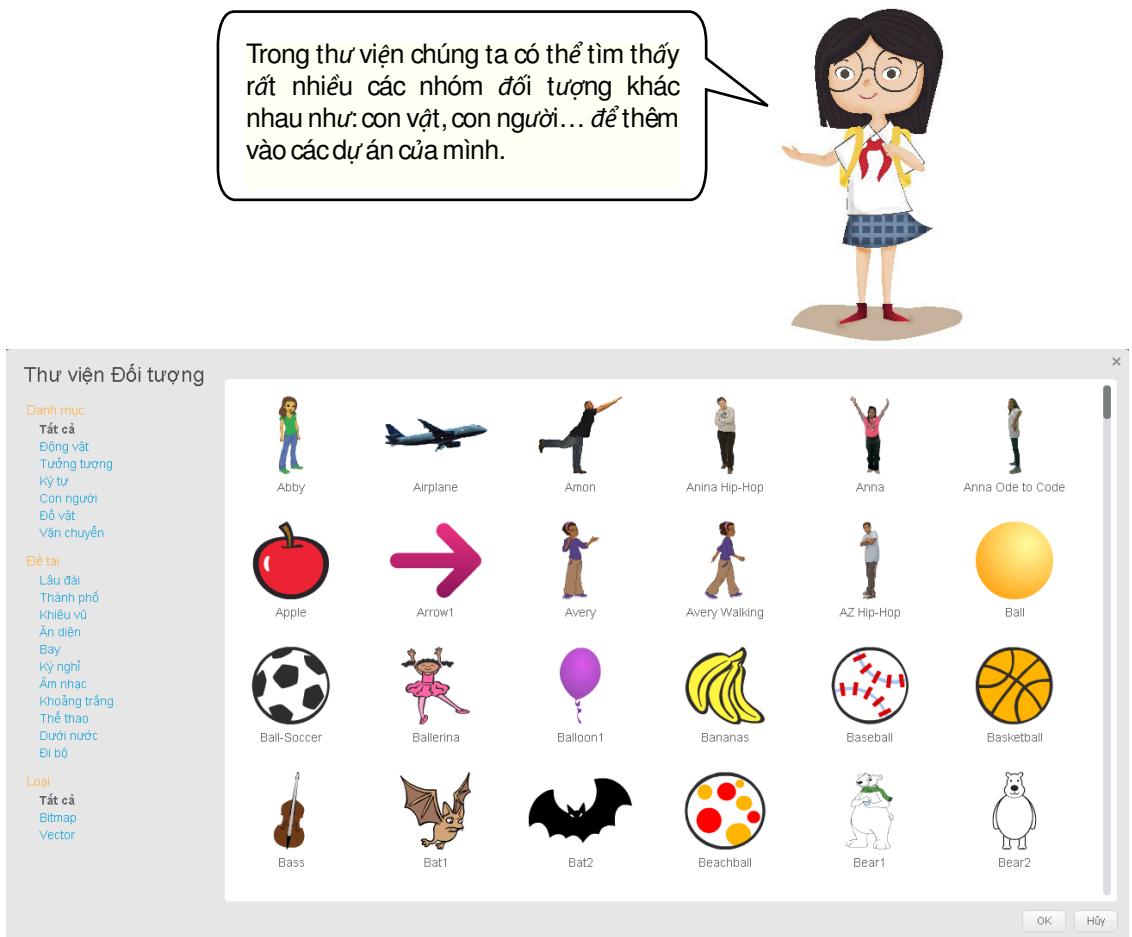
2. Giao diện

Giao diện Scratch được chia thành 2 phần cơ bản: Phần thể hiện dự án và phần phát triển dự án. Góc trên bên trái của giao diện Scratch là sân khấu (Stage). Sân khấu là nơi thể hiện dự án đang được thực thi. Khi một người chơi một trò chơi nào đó, sân khấu (Stage) chính là cửa sổ đang chạy chương trình trò chơi đó.

Theo mặc định, khi chương trình Scratch được khởi động thì biểu tượng **Scratch Cat** (chú mèo màu vàng cam) đã nằm ngay trên sân khấu. Biểu tượng **Scratch Cat** là một đối tượng đơn giản trong số rất nhiều đối tượng được Scratch xây dựng sẵn trong thư viện.

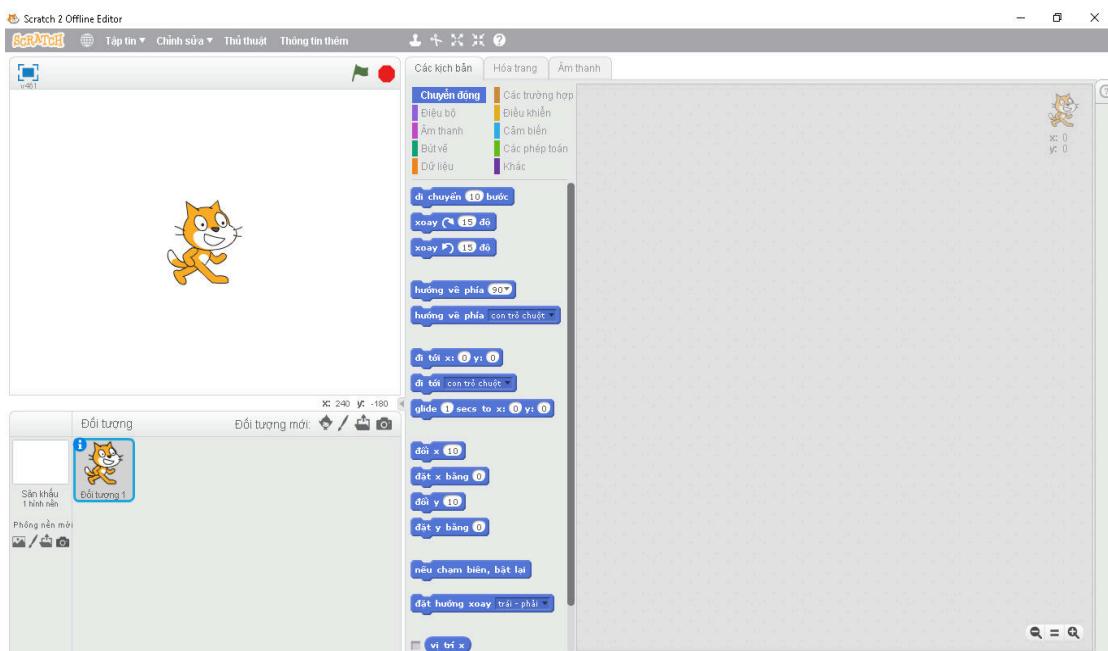
Để truy cập vào thư viện đối tượng trong Scratch, bạn nháy chuột chọn biểu tượng nhân vật tại khung Đối tượng mới.



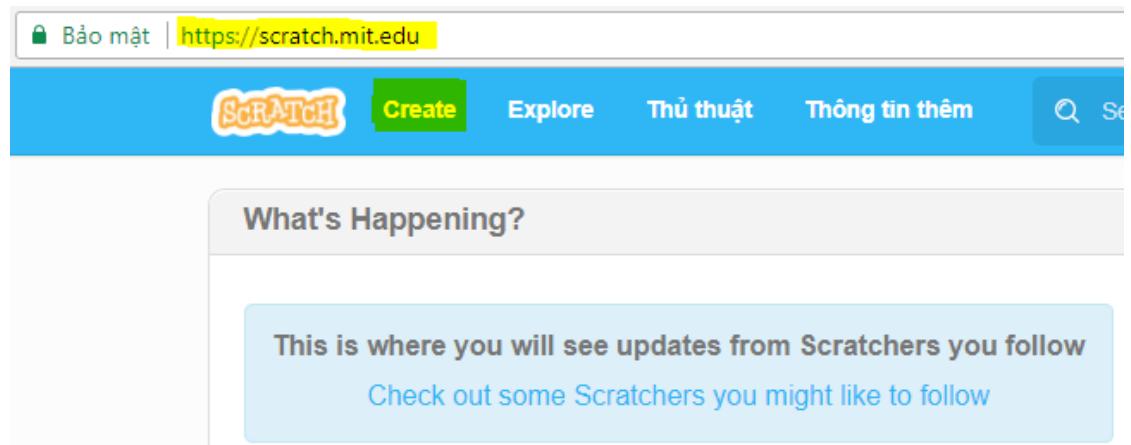


Hình: Thư viện đối tượng của Scratch

Đối tượng có thể được lập trình để thực hiện bất cứ điều gì bạn muốn. Tính linh hoạt của Scratch cho phép người lập trình hiện thực hóa trí tưởng tượng của mình thành các dự án. Với Scratch, dường như người lập trình không bị giới hạn về bất cứ thứ gì, chỉ cần có ý tưởng sáng tạo là bạn có thể hình thành dự án cho riêng mình.



Đối với bản Scratch sử dụng trực tuyến, các bạn cần truy cập vào địa chỉ trang web: <http://scratch.mit.edu> sau đó bấm chọn **Create** để tạo mới một dự án, và các bạn sẽ thấy giao diện giống hệt như trên.



Hình: Sử dụng Scratch online

3. Lập trình

Trước khi tìm hiểu thêm về giao diện Scratch, cách nhanh nhất để bạn có thể hiểu được một đối tượng (Sprite) đã được lập trình như thế nào là thử nghiệm những lệnh trong Scratch. Thực hiện theo các bước sau đây khi chương trình Scratch được khởi động và bạn chưa thực hiện một dự án nào.

Bước 1: Truy cập vào vùng **Các kịch bản (Scripts)** của chương trình Scratch.



Trong phần **Các kịch bản** bạn sẽ nhìn thấy có tất cả 10 nhóm lệnh được phân biệt bởi các màu sắc và nhóm chức năng khác nhau:

NHÓM LỆNH	MÔ TẢ CHỨC NĂNG
Chuyển động	Mô tả chuyển động của đối tượng
Điệu bộ	Các hiệu bộ, hình dạng của đối tượng
Âm thanh	Điều khiển âm thanh
Bút vẽ	Điều khiển bút và màu sắc nét vẽ
Dữ liệu	Điều khiển và quản lý dữ liệu
Các trường hợp	Các trường hợp có thể xảy ra như một điều kiện cho các hoạt động của đối tượng
Điều khiển	Điều khiển đối tượng
Cảm biến	Cảm biến va chạm, màu sắc, độ ồn, vị trí...
Các phép toán	Quản lý phép toán, biểu thức logic, ...
Khác	Tạo một khối lệnh mới hoặc thêm phần mở rộng

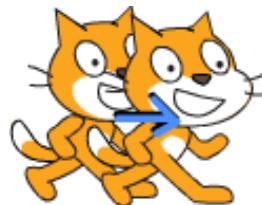
Bước 2: Nháy chuột chọn lệnh màu xanh có tên là **Di chuyển ()** **Bước** và kéo nó sang khung bên phải của màn hình.



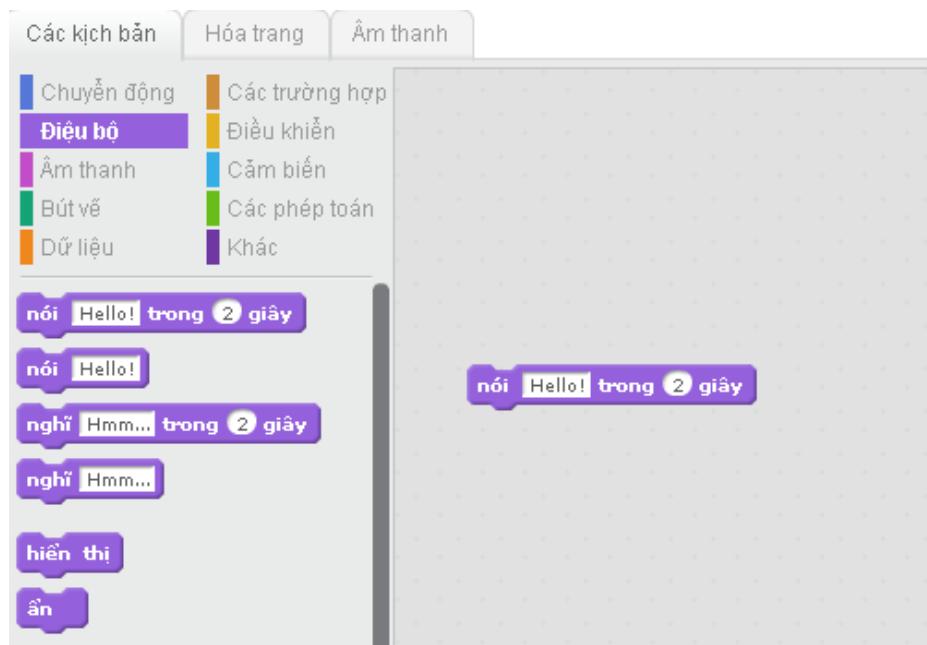
Bước 3: Thả chuột để đặt lệnh vừa kéo; đảm bảo rằng lệnh của bạn đã được đặt trong khung màu xám bên phải (khu vực này được gọi là khu vực kịch bản hay khu vực mã lệnh).



Bước 4: Sau khi hoàn tất, hãy Nháy chuột vào lệnh màu xanh **di chuyển 10 bước** và quan sát kỹ điều gì xảy ra với con mèo **Scratch**... Nó sẽ di chuyển 10 bước về phía trước (từ trái qua phải).



Bước 5: Nháy chuột ra ngoài và kiểm tra những lệnh khác để xem hoạt động của chúng như thế nào.



4. Khối lệnh (Block)

Block là những khối lệnh được xây dựng sẵn để thực hiện các dự án Scratch bằng cách lập trình kéo thả. Chúng có những chức năng xác định và khác biệt với nhau. Một vài lệnh có thể sẽ nằm trong các lệnh khác như các hình ví dụ dưới đây:

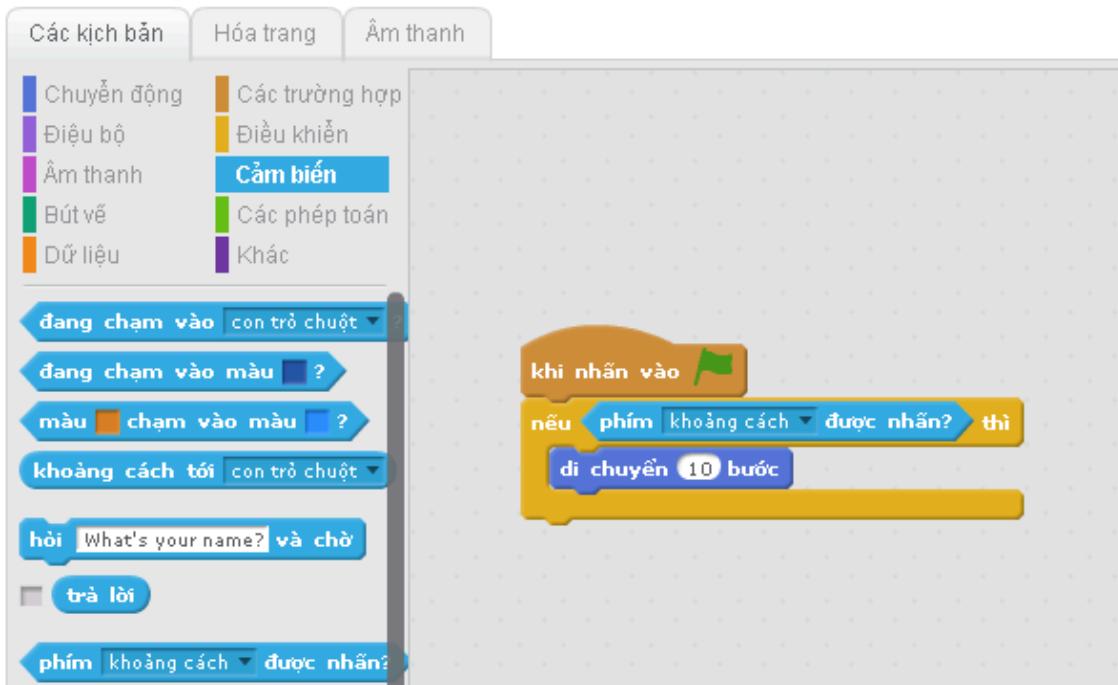
a. Lắp ráp các lệnh dưới đây. Lưu ý, khi truy cập vào các lệnh khác nhau, chúng ta nên chú ý đến màu sắc tương ứng của các thể loại để lựa chọn phù hợp.



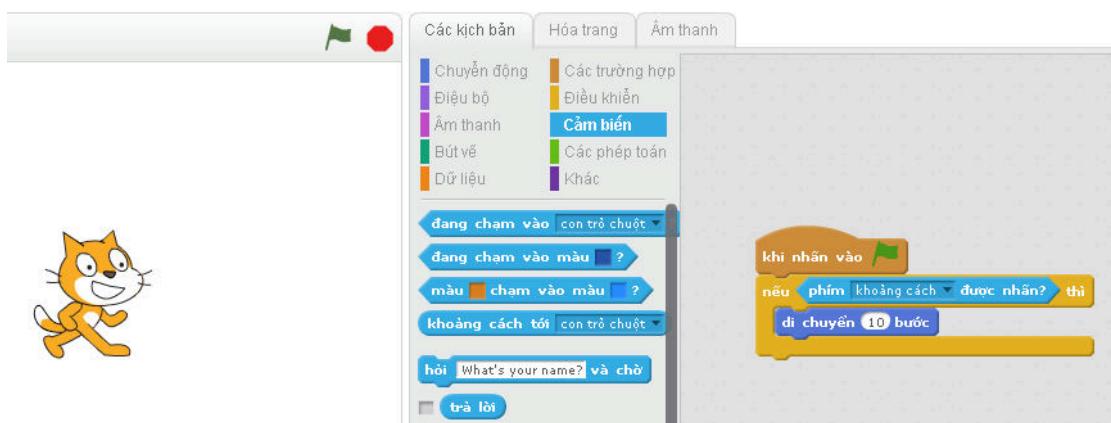
b. Ghép các lệnh vào theo hình sau:



c. Ghép lệnh màu xanh (**phím [khoảng cách] được nhấn?**) vào ô trống hình lục giác của lệnh **Nếu...thì** màu vàng cam:



d. Nháy chuột vào lá cờ màu xanh để chạy chương trình, và xem điều gì sẽ xảy ra.



e. Khi nháy chuột vào lá cờ và sau đó nhấn phím cách (Space) trên bàn phím, bạn không thấy có điều gì xảy ra hết. Tại sao vậy? Hãy nhìn kỹ lệnh chúng ta đã lắp ráp một lần nữa, và nhớ rằng một kịch bản (Script) phải thể hiện được sự liên kết giữa các lệnh một cách hợp lý.



Dự án bắt đầu chạy khi chúng ta nháy chuột vào biểu tượng lá cờ màu xanh lá cây, do đó lệnh tiếp theo đã được thực hiện. Và khi kích bản chạy, chương trình kiểm tra thấy phím cách chưa được nhấn và lệnh di chuyển 10 bước không được thực hiện.

Bây giờ, hãy khởi động lại dự án một lần nữa, nhưng lần này khi bắt đầu chạy dự án bằng cách nháy chuột vào lá cờ màu xanh, hãy nhấn giữ phím cách, các bạn sẽ thấy đối tượng của chúng ta di chuyển 10 bước.

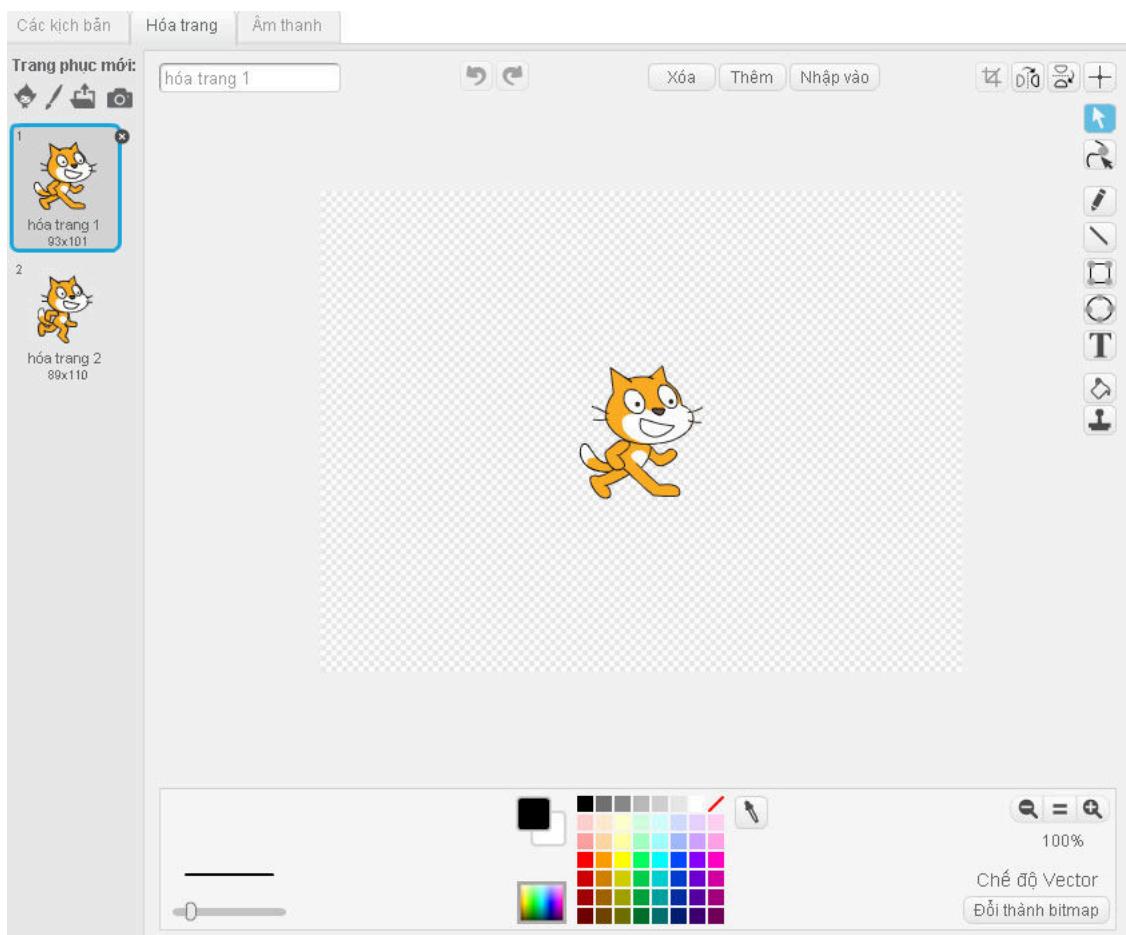
5. Biên tập âm thanh và hình vẽ

Scratch không chỉ có những mã lệnh được mô đun hoá mà còn được trang bị một trình biên tập về hình vẽ và âm thanh rất riêng.

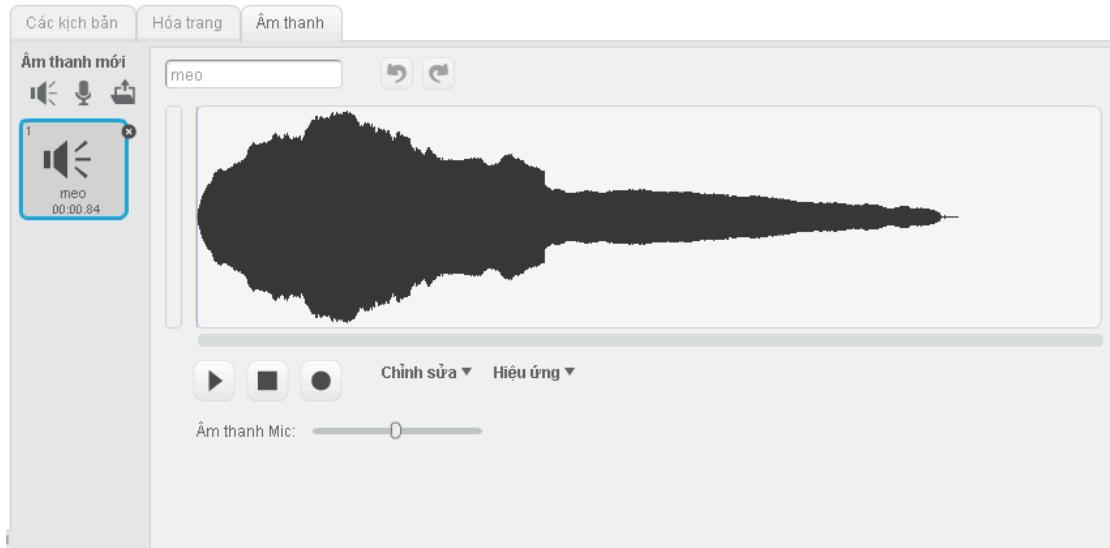
- Trình biên tập hình vẽ là chương trình dùng để thiết kế và chỉnh sửa các hình ảnh nhằm tạo ra những đối tượng (đối tượng, hình ảnh,...) mang cá tính riêng.

- Trình biên tập âm thanh cho phép chọn âm thanh từ thư viện, thu âm thanh mới, tải âm thanh từ máy tính và chỉnh sửa âm thanh được sử dụng trong một dự án Scratch.

Để truy cập nhanh vào 2 trình biên tập này, chúng ta chọn **Hóa trang** (trình biên tập hình vẽ) và **Âm thanh** (trình biên tập âm thanh).



Hình: Trình biên tập hình vẽ



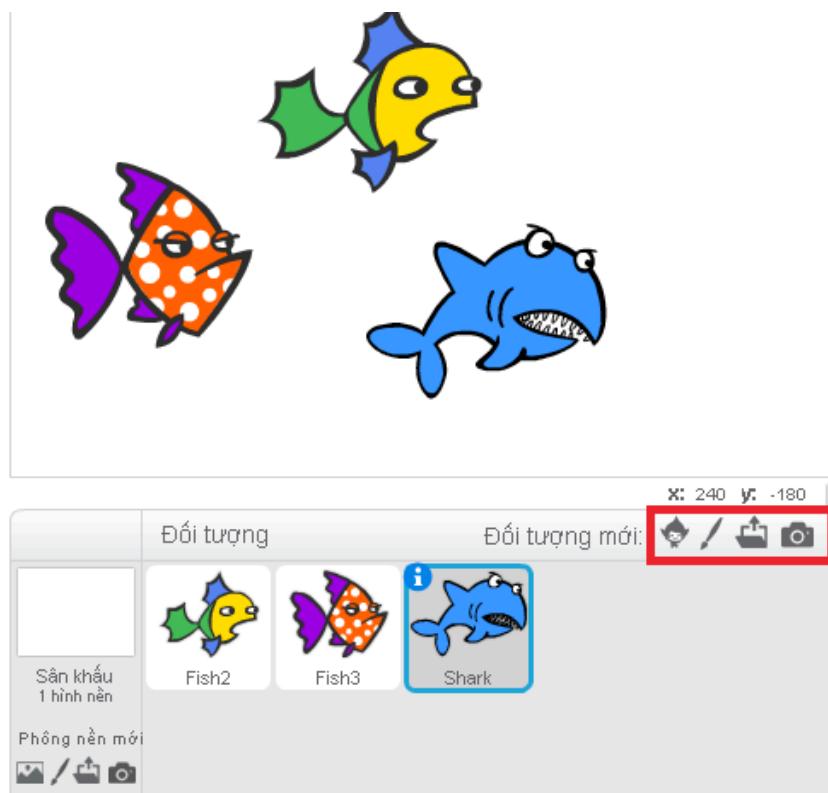
Hình: Trình biên tập âm thanh

6. Khung đối tượng

Trong Scratch, không phải tất cả các đối tượng (Sprites) đều thực hiện các chức năng tương tự nhau trong một dự án nhưng chúng được lập trình rất khác nhau về sự xuất hiện trên sân khấu hay về kịch bản hoạt động và âm thanh.

Chúng ta thực hiện truy cập vào các đối tượng khác nhau trong khung đối tượng nằm phía dưới phần sân khấu thực thi. Đối tượng đang được lựa chọn là những đối tượng có một hộp màu xanh bao quanh nó. Khi đó, bạn có thể truy cập tất cả các dữ liệu của đối tượng này.

Dưới đây là hình ảnh khung chứa các đối tượng trong một dự án Scratch:



7. Tạo mới một đối tượng



Hầu hết các dự án của Scratch đều có nhiều hơn một đối tượng. Vậy làm thế nào để tạo một đối tượng mới trong Scratch?

Quan sát kỹ trong khung đối tượng, chúng ta thấy có 4 nút dùng để tạo ra một đối tượng mới.



Với các nút này, một đối tượng có thể được đưa vào dự án từ thư viện có sẵn trong Scratch hoặc thêm các đối tượng mới. Chức năng của các nút lệnh đi từ trái sang phải như sau:

-  **Chọn đối tượng từ thư viện:** Chọn đối tượng từ thư viện đã được Scratch xây dựng sẵn với nhiều nhóm đối tượng khác nhau.
-  **Vẽ đối tượng mới:** Tạo mới một đối tượng bằng trình biên tập hình vẽ của Scratch.
-  **Tải lên đối tượng từ tệp tin:** Tạo mới một đối tượng bằng cách tải các hình vẽ hoặc hình ảnh từ máy tính của bạn vào dự án Scratch.
-  **Đối tượng mới từ máy ảnh:** Tạo mới một đối tượng bằng cách sử dụng camera của máy tính để chụp lại hình ảnh và thêm vào dự án Scratch.

8. Phần mở rộng – Thực hành

Một trong những cách nhanh nhất để có thể nâng cao kiến thức khi mới bắt đầu làm quen với Scratch là tự tìm tòi khám phá, chơi và trải nghiệm. Cố gắng tạo ra các đoạn lệnh khác nhau cho đối tượng, tìm hiểu về trình soạn biên tập hình vẽ, và các tính năng trong trình biên tập âm thanh. Điều này sẽ giúp bạn nhanh chóng hiểu sâu hơn về Scratch. Các bạn cũng có thể tận dụng các nguồn tài nguyên có sẵn trên Internet như các diễn đàn, Wiki Scratch để khám phá khi cần thiết.



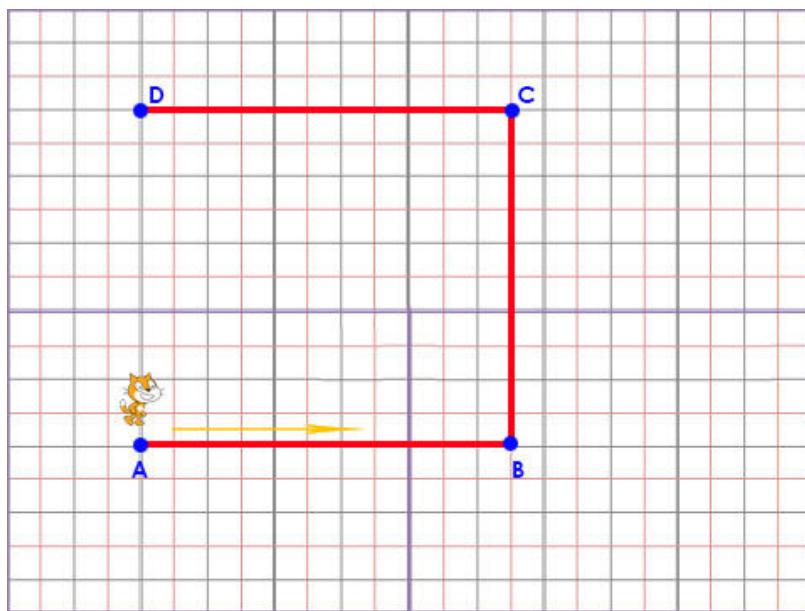
Mô đun 2. BÀI TOÁN CHUYỂN ĐỘNG

MỤC TIÊU

- Khám phá các nhóm lệnh chuyển động.
- Hiểu được cách thức di chuyển và định vị của đối tượng.
- Sử dụng các lệnh di chuyển, xoay và đổi hướng một cách phù hợp.
- Định vị cho đối tượng phù hợp trong các kịch bản khác nhau.
- Thay đổi trang phục đối tượng phù hợp với từng kịch bản.
- Thiết kế được một đoạn phim hoạt hình hoặc một trò chơi chuyển động đơn giản.
- Thay đổi hoạt cảnh một cách hợp lý.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

Quan sát hình vẽ sau:



Lưu ý: Mỗi ô vuông tương ứng với 20 pixel (20 bước).

1. Trả lời các câu hỏi

- Để di chuyển từ điểm A đến điểm D theo đường màu đỏ và hướng mũi tên vàng, chú mèo sẽ trải qua những hoạt động gì?
 - Để di chuyển từ điểm A đến điểm B, chú mèo sẽ phải di chuyển bao nhiêu bước, biết rằng mỗi ô vuông tương ứng với 20 bước?

- Làm thế nào để chú mèo di chuyển được từ điểm B đến điểm C và từ điểm C đến điểm D?

2. Tìm hiểu các nhóm lệnh

2.1. Nhóm lệnh chuyển động

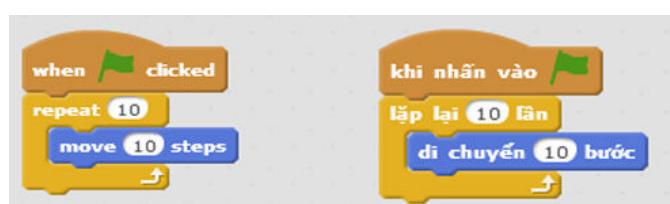
Nhóm lệnh chuyển động được sử dụng để biểu thị hành động của đối tượng. Nó có thể dễ dàng áp dụng cho những loại chuyển động khác nhau trong Scratch. Hầu hết các dự án Scratch đều có chuyển động. Hãy tìm hiểu, khám phá chức năng của các lệnh sau. Dựa vào phần ví dụ, gợi ý của từng lệnh, ghi chức năng của chúng ở bên dưới của mỗi lệnh.



a. Lệnh Di chuyển () Bước ()/ Move () Steps

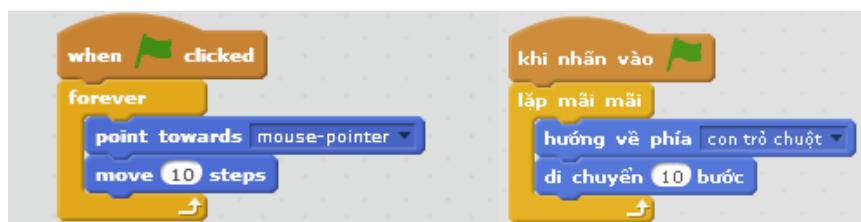
Một vài ví dụ thường dùng Di chuyển () Bước.

- Tạo di chuyển cho đối tượng - Di chuyển về phía trước với số lần lặp xác định.



Trong ví dụ này: Đối tượng của chúng ta sẽ di chuyển 10 lần (vòng lặp 10 lần – repeat 10), mỗi lần đối tượng **di chuyển 10 bước**.

- Làm cho đối tượng di chuyển theo chuột (Mouse).

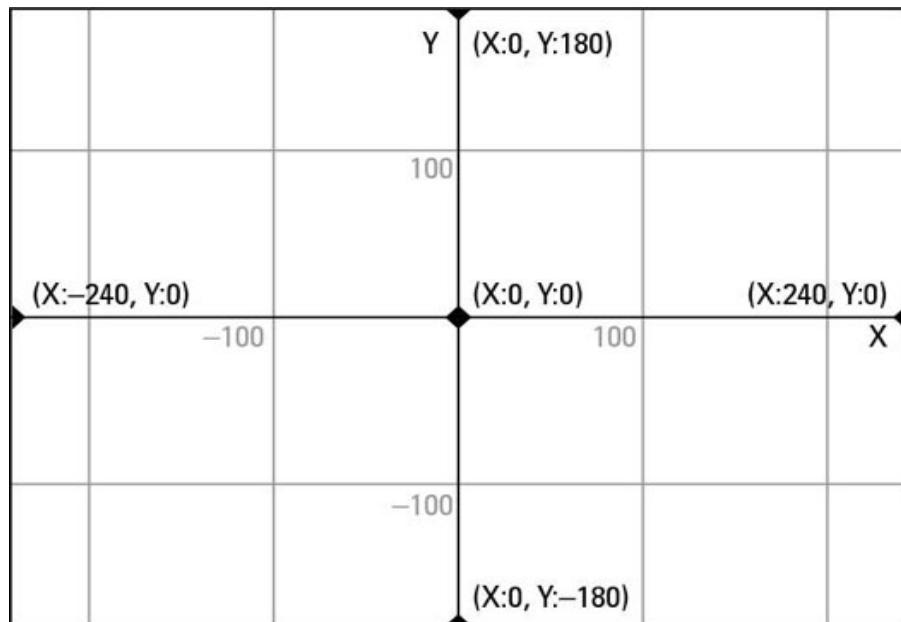


Trong ví dụ này: Đối tượng sẽ di chuyển liên tục (trong vòng **lặp mãi mãi**) theo hướng của con chuột máy tính, mỗi lần di chuyển 10 bước (tương ứng với 10 pixel).

Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Di chuyển () Bước**.

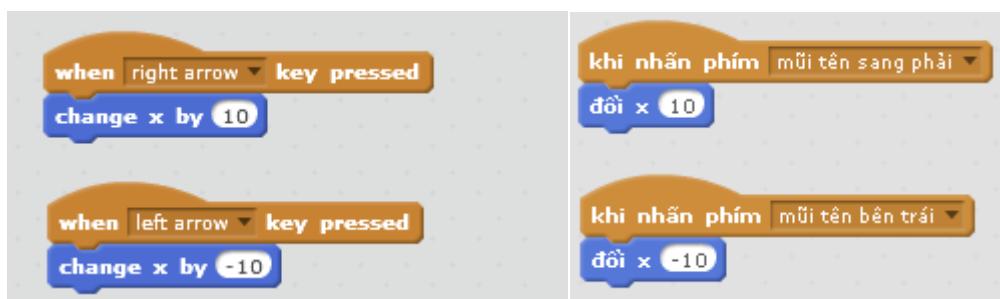
b. **Đổi X ()/Change X by ()**

Quan sát trực tiếp độ trên màn hình Scratch và ghép nối các lệnh: **Đi lên (tăng y), đi xuống (giảm y), qua phải (tăng x), qua trái (giảm x)**



Các ví dụ thường dùng **Đổi X ()**:

- Thông thường trong các trò chơi, người chơi sẽ điều khiển đối tượng của mình di chuyển qua lại, do đó lệnh **Đổi X ()** và **Đổi Y ()** là những lệnh phổ biến.



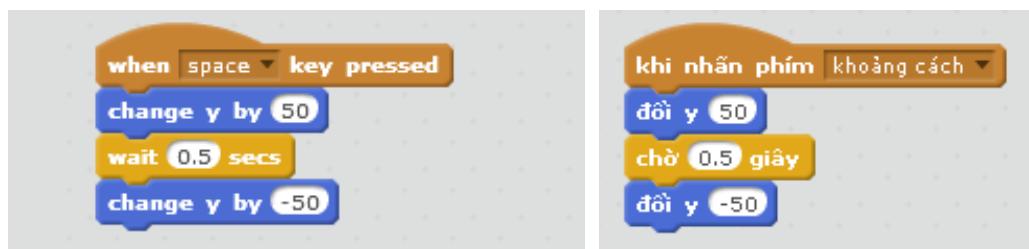
Trong ví dụ này, đối tượng của chúng ta sẽ di chuyển từ trái qua phải khi ta nhấn phím mũi tên phải (tọa độ X của đối tượng được thay đổi một lượng bằng 10) và di chuyển từ phải qua trái khi ta nhấn phím mũi tên trái (tọa độ X của đối tượng được thay đổi một lượng bằng -10).

Lưu ý: Trong các di chuyển của đối tượng, theo mặc định, nếu chuyển động từ trái sang phải thì các giá trị di chuyển sẽ mang giá trị dương; còn nếu đối tượng di chuyển từ phải qua trái thì các giá trị di chuyển sẽ mang giá trị âm (thêm dấu trừ vào trước giá trị).

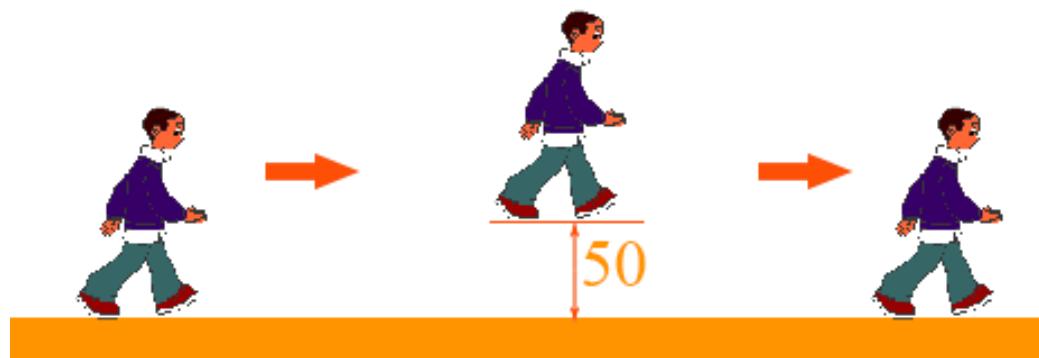
Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi X ()**

c. Lệnh Đổi Y ()/ Change Y by ()

Ví dụ sử dụng **Đổi Y ()**:



Trong ví dụ này, đối tượng của chúng ta sẽ nhảy lên một lượng là 50 pixel (thay đổi Y một lượng bằng 50) khi ta nhấn phím cách và sau 0,5 giây, đối tượng sẽ tự động rơi xuống một lượng là 50 pixel (thay đổi Y một lượng bằng -50).



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi Y ()**

d. Lệnh Xoay () độ/Turn () degrees

Một vài ví dụ phổ biến sử dụng Xoay () độ

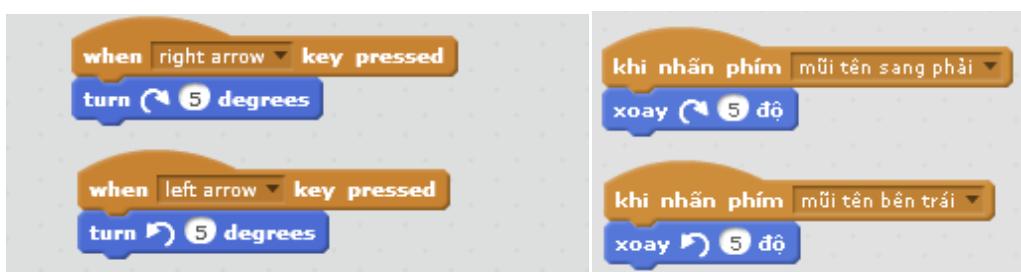
- Xoay đơn giản – Bánh xe quay



- Chuyển động – Vẫy tay



- Điều khiển phương tiện hoặc đối tượng xoay theo hướng xác định



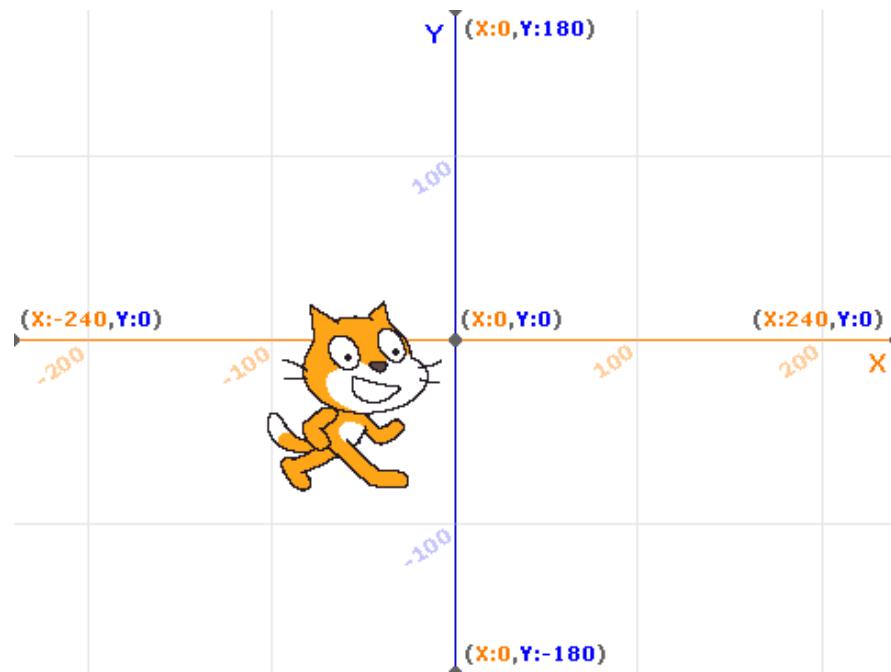
Trong ví dụ này, đối tượng sẽ xoay theo chiều kim đồng hồ 5 độ khi chúng ta nhấn phím mũi tên phải, và xoay ngược chiều kim đồng hồ khi chúng ta nhấn phím mũi tên trái.

Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Xoay () độ**.

2.2. Nhóm lệnh trượt, thay đổi vị trí và định vị

Trong phần này chúng ta sẽ tìm hiểu về các lệnh cho phép xác định vị trí tức thời hoặc vị trí của đối tượng trong suốt quá trình chuyển động của chúng trong dự án bằng việc sử dụng hệ toạ độ X, Y. Các vị trí đó có thể được xác định bởi một tập hợp các toạ độ X, Y, vị trí của con trỏ chuột hoặc của một đối tượng bất kỳ nào đó trong dự án. Dựa vào hệ toạ độ dưới đây và các ví dụ minh họa cho từng lệnh, hãy tự nêu chức năng của chúng ở cuối mỗi phần trình bày.





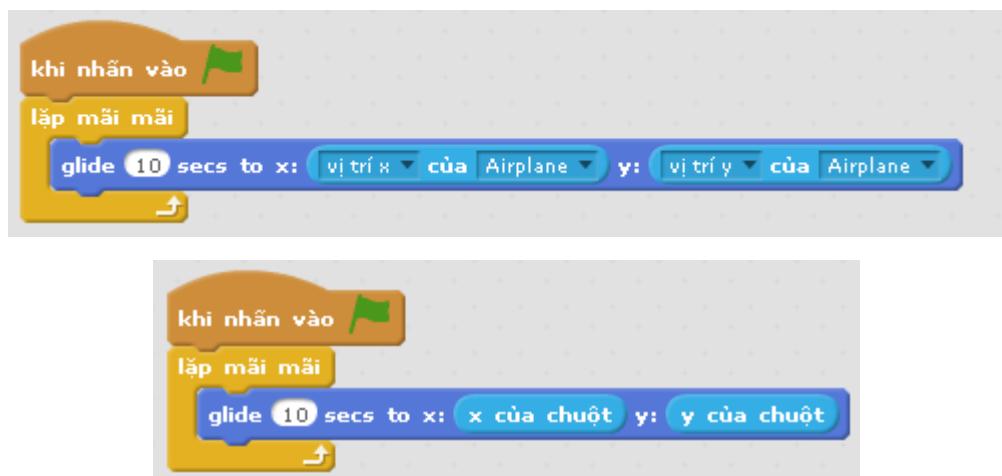
a. Lệnh Glide () Secs to X: () Y: ()

Một vài ví dụ phổ biến sử dụng lệnh Glide () Secs to X: () Y: ()

- Đối tượng trượt nhẹ tới một vị trí xác định.



- Đối tượng lướt nhẹ tới vị trí (tọa độ) của một đối tượng khác hoặc vị trí con trỏ chuột.



- Đối tượng rơi xuống.



- Tạo các chướng ngại vật trượt ngược chiều với đối tượng.



Từ quan sát các ví dụ trên, hãy nêu chức năng của lệnh **Glide () Secs to X: () Y: ()**

So sánh giữa lệnh Glide () và Di chuyển () Bước

- Kịch bản này sử dụng lệnh **Di chuyển () Bước** để di chuyển đối tượng từ cạnh trái sang cạnh phải của sân khấu trong thời gian 8 giây.



- Kịch bản này thực hiện các hoạt động tương tự như sử dụng lệnh **Glide ()**.



Hãy thực hiện lại, quan sát kỹ và cho biết cảm nhận của bạn về 2 kịch bản trên.

b. Lệnh Di tới X () Y ()

Một vài ví dụ về lệnh Di tới X () Y ()

Trong các dự án Scratch thì đối tượng thường phải nhảy tới một vị trí nào đó trên sân khấu – và việc sử dụng lệnh **Đi tới X () Y ()** chính là cách đơn giản nhất để thực hiện việc đó. Sau đây là một vài trường hợp phổ biến sử dụng lệnh này trong dự án Scratch.

- Di chuyển đối tượng tới vị trí xác định.



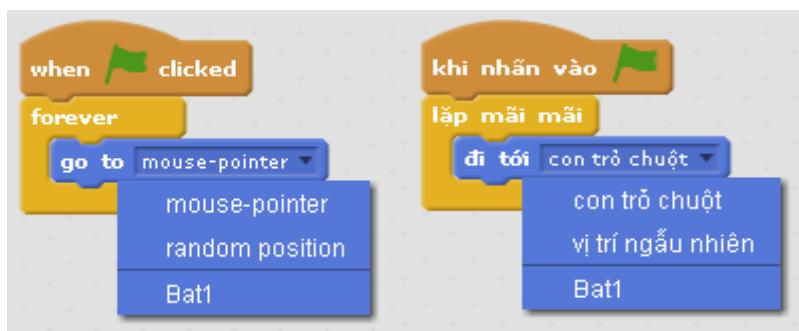
- Thiết lập lại vị trí ban đầu của đối tượng khi bắt đầu chạy dự án.



- Di chuyển đối tượng tới một vị trí để in hình đối tượng đó trên sân khấu.



- Di chuyển đối tượng tới vị trí của con trỏ chuột, vị trí ngẫu nhiên hoặc một Sprite khác trong dự án.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đi tới X () Y ()**

c. Lệnh Đặt X bằng ()/ Set X to () và Đặt Y bằng ()/ Set Y to ()

Một số ví dụ sử dụng lệnh Đặt X bằng () – Đặt Y bằng ():

Nếu Y (hoặc X) của một đối tượng trong dự án được giữ nguyên không thay đổi trong một hoạt động nào đó nhưng vẫn cần có sự di chuyển của đối tượng (ví dụ một thanh cuộn ngang hoặc dọc), lệnh **Đặt X bằng ()** (hoặc **Đặt Y bằng ()**) được sử dụng thay cho lệnh **Đi tới X () Y ()**.

Lệnh này thường được sử dụng trong các dự án có sự mô tả sự dịch chuyển của cảnh quan như có một thanh cuộn trên sân khấu chính của Scratch.

- Thiết lập vị trí của đối tượng một cách ngẫu nhiên trên màn hình.



Thiết lập vị trí ngẫu nhiên cho đối tượng để lấp đầy màn hình như một dự án có sự xuất hiện của bông tuyết.



- Thiết lập lại giá trị X của một thanh trượt.



- Di chuyển vợt bóng bàn trong một trò chơi đánh bóng bàn.

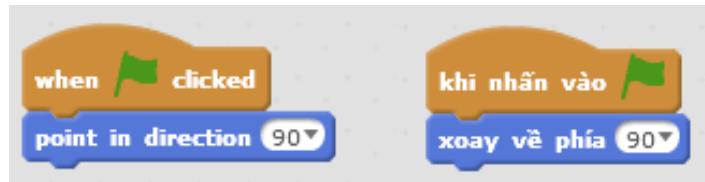


Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đặt X bằng () – Đặt Y bằng ()**.

d. Lệnh Hướng về phía () / Point in Direction () – Point towards ()

Một vài ví dụ phổ biến của việc sử dụng lệnh này.

- Thiết lập lại hướng ban đầu cho đối tượng khi dự án bắt đầu chạy.



- Lật ngược một đối tượng:



- Hướng đối tượng theo con trỏ chuột hoặc một đối tượng khác trong dự án:



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Hướng về phía ()**.

2.3. Thay đổi trang phục và nền sân khấu

Trong phần này, chúng ta sẽ tìm hiểu về cách lựa chọn ngoại hình trang phục cho đối tượng và cách thay đổi hình nền sân khấu cho phù hợp với các đoạn kịch bản khác nhau. Quan sát và thực hiện lại các ví dụ và gợi ý trong từng lệnh; sau đó đưa ra nhận xét, chức năng của từng lệnh ở cuối mỗi phần trình bày.

a. Đổi hình dạng thành ()/ Switch Costume to ()

Một số ví dụ áp dụng lệnh **Đổi hình dạng thành ()** trong dự án Scratch.

Lệnh **Đổi hình dạng thành ()** hoặc **Đổi hình nền thành ()** có thể được sử dụng trong việc tạo các đoạn phim hoạt hình, trò chơi, mô phỏng, hoặc bất cứ hoạt động nào đòi hỏi việc thay đổi trang phục cho các đối tượng.

- Thay đổi thành trang phục, ngoại hình khác:



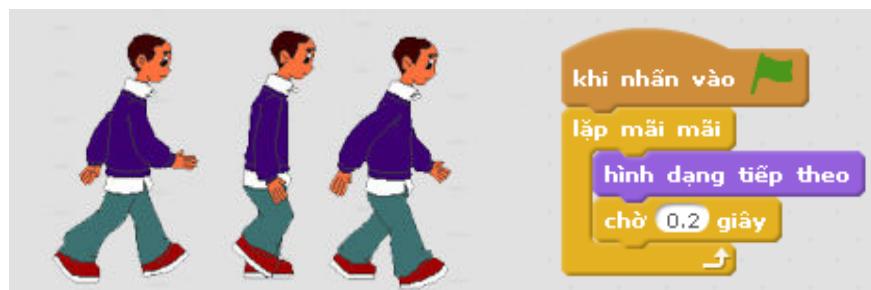
- Thay đổi một trang phục như một phần của 1 chuỗi liên tục:



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi hình dạng thành ()**.

b. Lệnh Hình dạng tiếp theo/ Next Costume

Lệnh này được dùng chủ yếu trong các đoạn kịch bản phim hoạt hình (animation), một đối tượng sẽ được thay đổi trang phục liên tục. Lệnh **Chờ () giây** thường được sử dụng cùng với **Hình dạng tiếp theo** để thay đổi tốc độ, giúp chúng ta điều chỉnh tốc độ chuyển đổi trang phục một cách phù hợp với kịch bản.



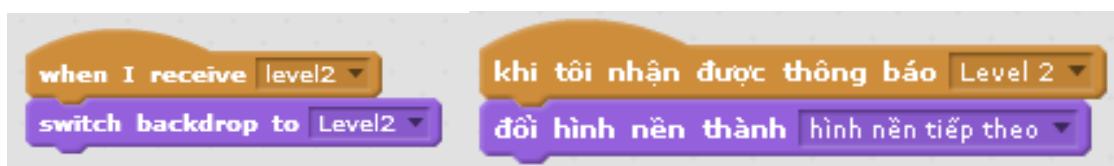
Từ quan sát ví dụ trên, hãy nêu chức năng của lệnh **Hình dạng tiếp theo**.

c. Lệnh Đổi hình nền thành ()/ Switch Backdrop to ()

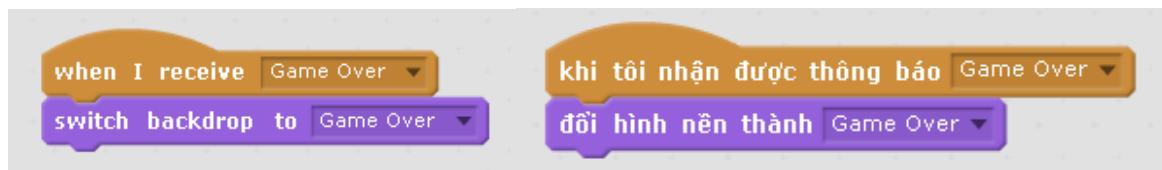
Lệnh này được sử dụng rất nhiều trong các dự án Scratch bởi tính hữu dụng và sự đơn giản của nó trong quá trình sử dụng.

Một vài ví dụ phổ biến sử dụng lệnh như sau:

- Khi bắt đầu một cấp (level) trong một dự án trò chơi.



- Khi thông báo trò chơi kết thúc.

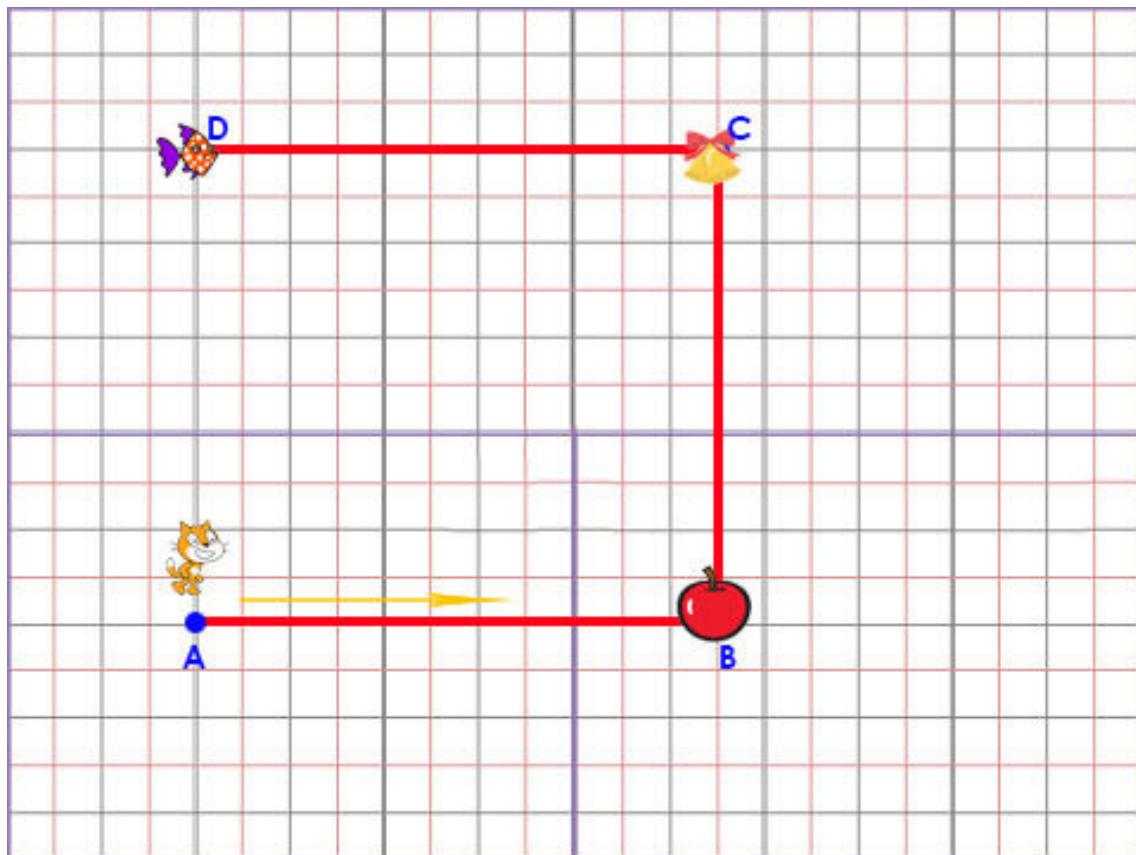


Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi hình nền thành ()**

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Quan sát lại hình vẽ ở phần A, lập trình mô phỏng lại hoạt động của chú mèo khi di chuyển từ điểm A đến điểm D theo chiều mũi tên và đường kẻ màu đỏ (sử dụng lệnh **Di chuyển () Bước** và các lệnh đã tìm hiểu, khám phá được). Lưu lại bài làm với tên “Di Chuyen 1”.

2. Thêm vào chương trình 3 đối tượng bất kỳ và thiết lập vị trí của chúng tương ứng với các điểm B, C, và D. Giả sử chú mèo đứng tại điểm A có tọa độ X= -160; Y= -80 (ví dụ như hình). Lưu lại bài với tên “Di Chuyen 2”.



3. Thực hiện lại bài “Di chuyen 1” bằng cách sử dụng lệnh **Glide () Secs to X () Y ()** và các lệnh đã tìm hiểu được ở phần trước. Lưu lại bài làm với tên là “Di Chuyen 3” sau đó so sánh hoạt động của đối tượng trong “Di Chuyen 1” và “Di Chuyen 3”.

4. Mở lại bài “Di Chuyen 2”, thực hiện thiết lập lại vị trí của Chú mèo là (-160; -80) khi bắt đầu chơi, sau đó cho chú mèo di chuyển theo hướng mũi tên và đường kẻ màu đỏ qua các điểm B, C và D (đến vị trí con cá). Khi chú mèo chạm đến điểm D thì chuyển nền Sân khấu sang một nền khác có chữ “Kết thúc hành trình”. Lưu lại bài làm với tên “Di Chuyen 4”.

C. HOẠT ĐỘNG SÁNG TẠO

Chia lớp học thành các nhóm khác nhau – Mỗi nhóm gồm 5 thành viên (bao gồm 1 trưởng nhóm) thực hiện việc xây dựng ý tưởng sáng tạo về một dự án trò chơi hoặc một đoạn phim hoạt hình ý nghĩa, phù hợp với lứa tuổi.

Yêu cầu bắt buộc:

- Dự án phải bao gồm ít nhất 2 đối tượng trở lên.
- Dự án phải có sự tương tác giữa các đối tượng, giữa người chơi và máy tính.
- Dự án phải áp dụng các kiến thức nắm được trong mô đun **Bài toán chuyển động**.
- Các đối tượng nên có các trang phục, hình dạng khác nhau để tạo sự chuyển động linh hoạt và đẹp mắt.
- Xây dựng kịch bản cho dự án.
- Phân tích các sự kiện, hành vi của đối tượng trong dự án.
- Phân công nhiệm vụ các thành viên trong nhóm (thiết kế đồ họa, lập trình, kiểm thử, sửa lỗi ...)
- Trình bày dự án.

Gợi ý 1: Hãy quan sát bể cá cảnh nhà bạn và mô phỏng lại những hoạt động của cá bằng các lệnh.

Gợi ý 2: Hãy mô phỏng lại bằng các lệnh một phần chuyển động của một trò chơi hay một đoạn phim hoạt hình nổi tiếng nào đó mà các bạn đã chơi hoặc đã xem và các bạn thấy thích thú – Ví dụ trò chơi Flappy Bird.

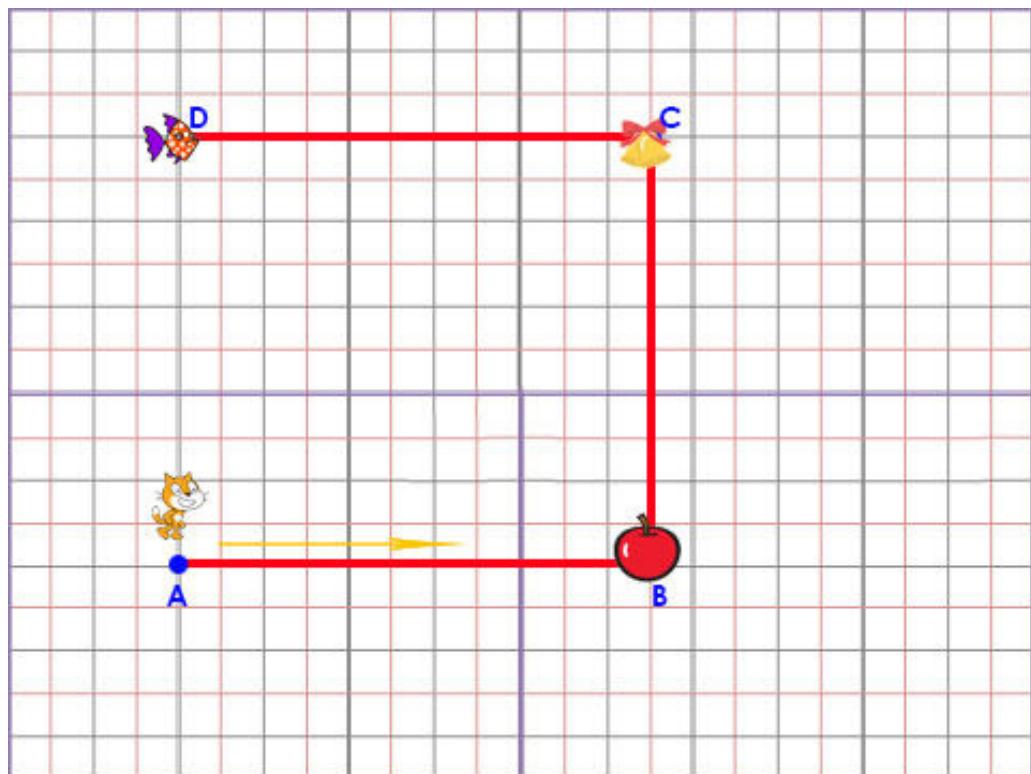
Mô đun 3. HỘI THOẠI VÀ HIỆU ỨNG ĐỒ HOẠ ĐỐI TƯỢNG

MỤC TIÊU

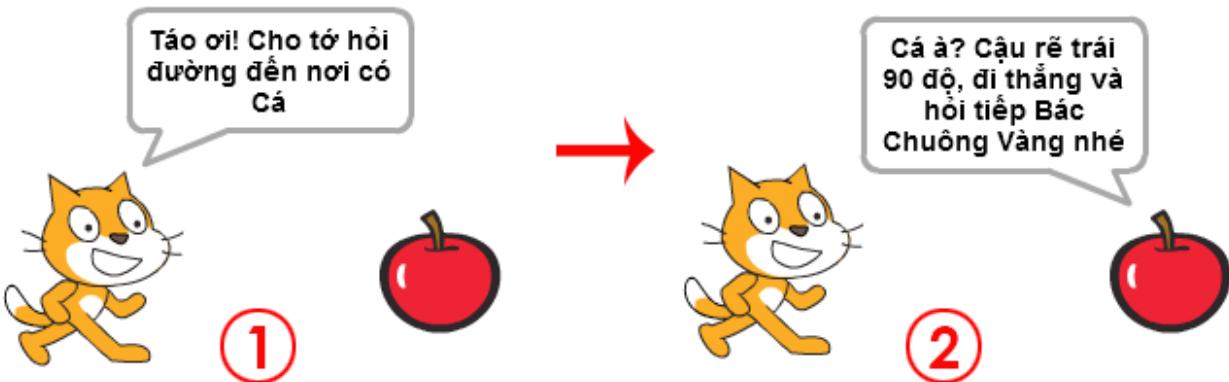
- Khám phá các hiệu ứng đồ hoạ đối tượng trong Scratch.
- Sử dụng hiệu ứng đồ hoạ một cách hợp lý.
- Thay đổi thứ tự các lớp đối tượng.
- Sử dụng các hoạt động chuyển cảnh nhằm gia tăng sự thú vị của trò chơi.
- Sử dụng các hiệu ứng đồ hoạ nhằm tăng thêm tính hấp dẫn, thu hút cho chương trình.
- Thiết kế được một đoạn phim hoạt hình hoặc một trò chơi chuyển động đơn giản.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Mở bài tập “Di Chuyen 4”



Hãy quan sát và trao đổi cùng các bạn của bạn để phát triển bài tập này thành một đoạn hội thoại “Hỏi đường đi tới vị trí con cá” của chú mèo với các đối tượng khác (quả táo và chuông vàng) trong bài “Di Chuyen 4”.



2. Trả lời câu hỏi

- Các đối tượng có thể trao đổi thông tin hay suy nghĩ với nhau được không?
- Làm thế nào để 2 đối tượng có thể trao đổi thông tin và nói chuyện với nhau như một cuộc hội thoại thông thường?
 - Làm thế nào để biểu thị suy nghĩ của đối tượng?
 - Làm sao để thay đổi màu sắc của quả táo khi nó chạm vào con mèo?
 - Làm thế nào để biểu thị rằng con mèo sẽ ăn con cá khi nó tìm thấy con cá?
 - Làm sao để sau khi ăn cá con mèo sẽ lớn lên một chút so với kích thước ban đầu?

3. Tìm hiểu các nhóm lệnh

Mô đun này giới thiệu về các khối lập trình cho phép chúng ta hiển thị một đoạn hội thoại, một câu nói hay một dòng biểu thị suy nghĩ với thông điệp tương ứng. Các thông điệp này được hiển thị vĩnh viễn hoặc trong thời gian xác định. Bên cạnh đó chúng ta cũng được tìm hiểu về các khối lập trình mà cho phép thay đổi các hiệu ứng đồ họa cho đối tượng. Những hiệu ứng đồ họa bao gồm: màu sắc, hiệu ứng mắt cá, Pixelate, độ sáng và bóng mờ...

Hãy quan sát các ví dụ và các gợi ý của từng lệnh, sau đó đưa ra chức năng của từng lệnh ở cuối mỗi phần trình bày.



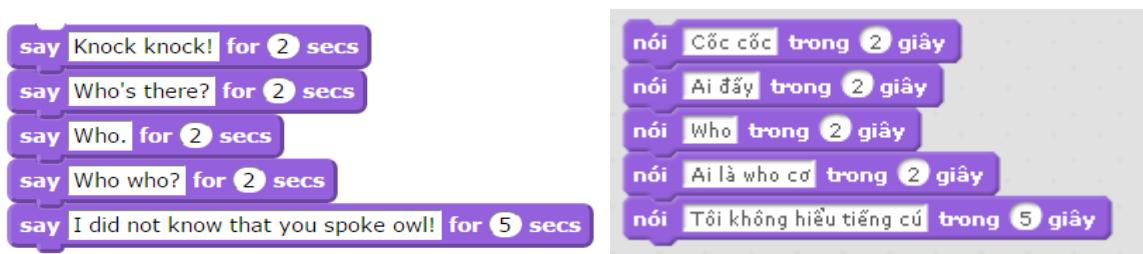
3.1. Đối tượng suy nghĩ và nói

a. Lệnh Nói () trong () giây / Say () for () Secs

Lệnh này có thể được sử dụng bất cứ khi nào bạn cần hiển thị một đoạn hội thoại cho đối tượng. Một trong những trường hợp phổ biến khi sử dụng lệnh này là hiển thị những hội thoại – văn bản không biết trước đã được lưu cho một biến nào đó của dự án.

Một vài ví dụ phổ biến sử dụng lệnh Nói () trong () giây:

- Hiển thị một đoạn hội thoại:



- Hiển thị một văn bản không biết trước:

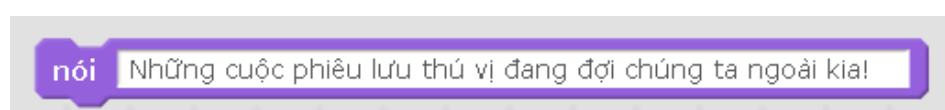


Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nói () trong () giây**.

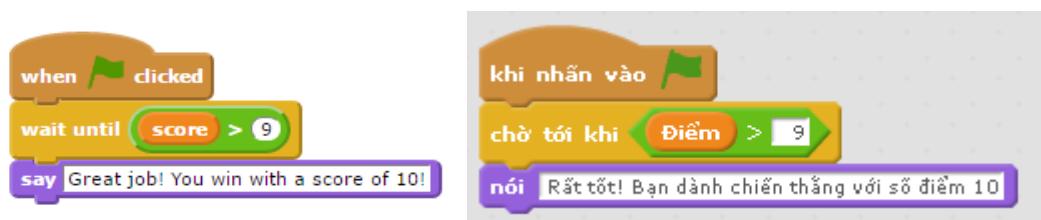
b. Lệnh Nói ()/ Say ()

Sau đây là một vài ví dụ phổ biến dùng lệnh Nói ():

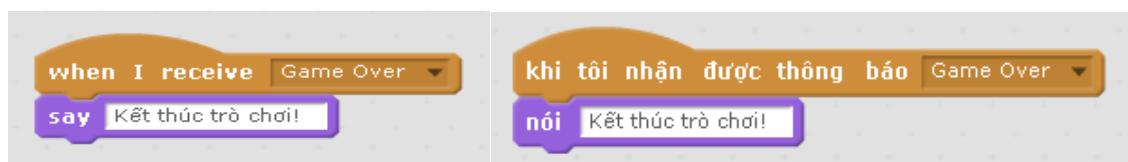
- Một thông điệp muốn hiển thị mãi mãi.



- Một thông điệp muốn hiển thị mãi mãi.



- Một thông báo kết thúc trò chơi.



- Kéo dài một đoạn hội thoại và chờ tác động của người chơi trước khi chuyển tới một lệnh khác.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nói ()**.

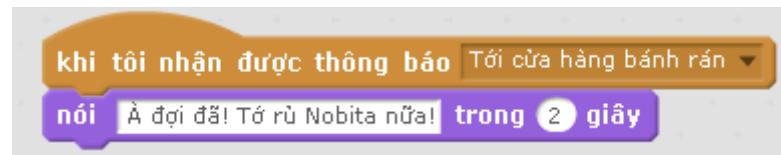
c. Lệnh Nghĩ () trong () giây / Think () for () Secs

Một vài ví dụ phổ biến sử dụng lệnh Nghĩ () trong () giây:

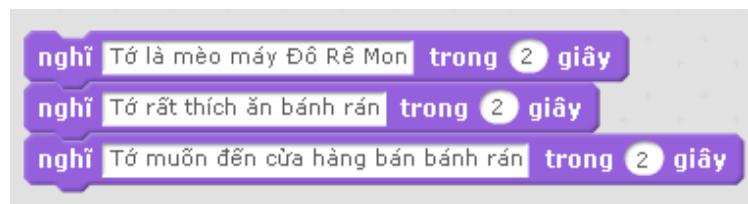
- Một cuộc hội thoại.



- Hiển thị một đoạn văn bản.



- Suy nghĩ của một đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nghĩ () trong () giây**.

d. Lệnh Nghĩ ()/ Think ()

Một vài ví dụ phổ biến sử dụng lệnh Nghĩ ():

- Một đối tượng luôn suy nghĩ.



- Hiển thị vĩnh viễn một suy nghĩ của đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nghĩ ()**.

3.2. Hiệu ứng đồ họa

Hiệu ứng đồ họa là dạng hiệu ứng được sử dụng trên đối tượng hay sân khấu, để làm thay đổi hình ảnh, màu sắc của đối tượng hoặc sân khấu. Chúng ta có thể tìm thấy các lệnh về đồ họa này trong nhóm lệnh **Điều bộ**.

Đây là 3 lệnh thường được dùng để thay đổi các hiệu ứng đồ họa trên đối tượng hoặc sân khấu:

- **Đổi hiệu ứng () một lượng () / Change () Effect by ()**
- **Đặt hiệu ứng () là () / Set () Effect to ()**
- **Xóa hiệu ứng đồ họa / Clear Graphic Effects**

a. Lệnh Đổi hiệu ứng () ()

Một vài ví dụ phổ biến sử dụng lệnh Đổi hiệu ứng () ()

- Thay đổi hiệu ứng màu sắc trên đối tượng:



- Sử dụng hiệu ứng gương cầu để tạo ra một đối tượng như thể đang được nhìn qua nước hoặc thấu kính góc rộng:



- Sử dụng hiệu ứng xoay tròn **Xoay** để làm biến dạng đối tượng:



- Làm cho các đối tượng có cảm giác nhìn trong suốt bằng hiệu ứng **Bóng ma**:



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi hiệu ứng () ()**.

b. Lệnh Đặt hiệu ứng () là () / Set () Effect to ()

Một vài ví dụ phổ biến dùng lệnh Đặt hiệu ứng () là ():

- Thay đổi màu sắc của một đối tượng.
- Dùng hiệu ứng **Gương cầu** để tạo cho đối tượng giống như nó đang ở trong nước hoặc đang được nhìn qua thấu kính góc rộng.
- Tạo sự khác biệt về cường độ sáng với hiệu ứng **Độ sáng**.



- Tạo sự trong suốt cho đối tượng với hiệu ứng **Bóng ma**.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đặt hiệu ứng () là ()**.

c. Lệnh Xóa hiệu ứng đồ họa/ Clear Graphic Effect

Lệnh này được dùng để xoá bỏ toàn bộ các hiệu ứng đồ họa trên đối tượng. Nó thường được dùng trong những trường hợp đối tượng phải thiết lập lại về hình dạng ban đầu của nó.

Một vài trường hợp phổ biến khi dùng Xóa hiệu ứng đồ họa

- Khi một đối tượng cần được xoá bỏ toàn bộ hiệu ứng để trở về trạng thái ban đầu.
- Khi thiết lập lại hình dáng ban đầu của một bức tranh.
- Khi một đối tượng, nhân vật bị biến dạng bởi tác động thay đổi hình dáng của nó.

d. Danh sách các loại hiệu ứng đồ họa

- **Màu (Color)** – Hiệu ứng màu sắc trên đối tượng.
- **Gương cầu (Fisheye)** – Hiệu ứng gương cầu, giống như chúng ta nhìn đối tượng thông qua một thấu kính góc rộng.
- **Xoay (Whirl)** – Hiệu ứng xoắn đối tượng quanh tâm của nó.
- **Nhòe (Pixelate)** – Hiệu ứng làm nhòe đối tượng.
- **Khảm (Mosaic)** – Hiệu ứng tạo ra nhiều đối tượng nhỏ từ một đối tượng ban đầu.
- **Độ sáng (Brightness)** – Thay đổi độ sáng trên đối tượng.
- **Bóng ma (Ghost)** – Thay đổi hiệu ứng mờ ảo trên đối tượng.

Một vài ví dụ phổ biến của các loại hiệu ứng đồ họa.

Màu sắc (Color)

- Tạo hiệu ứng kiểu nhấp nháy cho đối tượng.
- Thiết lập bầu không khí ở những nơi nhất định trong dự án.
- Tạo hiệu ứng khác nhau cho mỗi đối tượng.
- Tạo nghệ thuật bằng cách in hình đối tượng với những màu sắc khác nhau.

Gương cầu (Fisheye)

- Làm biến dạng đối tượng.
- Để tạo sự xuất hiện của đối tượng như thể nó đang tiến tới hoặc đi từ màn hình.

- Để tạo một đối tượng với hình dạng béo mập hơn.
- Có thể sử dụng nó như một sự chuyển tiếp các trang phục cho đối tượng.
- Để làm cho đối tượng như bị bóp méo, sứt mẻ.
- Để tượng trưng cho một thứ gì đó kỳ dị khác thường.

Xoay (Whirl)

- Thực hiện việc bóp méo, làm biến dạng một đối tượng.
- Để thể hiện đối tượng như đang bị quay tròn.
- Cũng có thể sử dụng như một sự chuyển tiếp trang phục cho đối tượng.

Nhở (Pixelate)

- Để thực hiện một dự án kiểu cổ xưa hoặc lỗi thời.
- Có thể sử dụng như một sự chuyển tiếp các trang phục cho đối tượng.

Độ sáng (Brightness)

- Để tạo một sự phát sáng ấn tượng cho đối tượng.
- Để làm một cái gì đó trong nền tối.

Bóng ma (Ghost)

- Tạo sự mờ ảo bên trong và bên ngoài.
- Làm cho một đối tượng nào đó biến mất.
- Trộn các màu sắc của các đối tượng.

3.3. Thay đổi kích thước đối tượng, thứ tự các lớp

Trong phần này, chúng ta sẽ tìm hiểu về các khối lập trình cho phép thiết lập hoặc thay đổi kích thước của đối tượng, làm cho nó lớn hơn hoặc nhỏ hơn so với kích thước ban đầu. Hãy quan sát các ví dụ và thực hiện lại chúng trên chính máy tính của mình, sau đó đưa ra nhận định về chức năng của từng lệnh ở cuối mỗi phần trình bày.

a. Lệnh Đổi kích thước một khoảng ()/ Change Size by ()

Ví dụ phổ biến khi sử dụng lệnh Đổi kích thước một khoảng ():

- Làm cho đối tượng lớn lên hoặc nhỏ đi khi thỏa mãn một điều kiện nào đó.



Từ quan sát ví dụ trên, hãy nêu chức năng của lệnh **Đổi kích thước một khoảng ()**.

b. Lệnh Đặt kích thước bằng () %/ Set Size to () %

Một số trường hợp phổ biến sử dụng lệnh Đặt kích thước bằng () %

- Làm tăng kích thước của một nút lệnh (button) khi đưa chuột qua nút đó.



- Làm đổi tương lớn lên theo tiến trình thời gian.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đặt kích thước bằng () %**

c. Lệnh Hiển thị - Ẩn/ Show - Hide

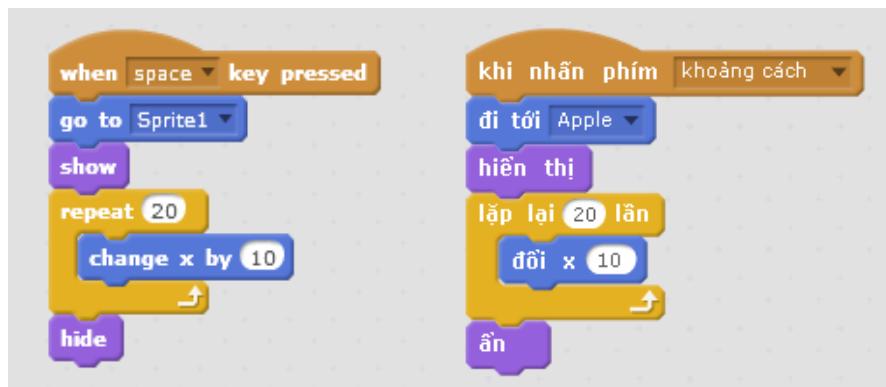
Hai lệnh **Hiển thị/ Ẩn** cũng là hai trong số những lệnh được dùng phổ biến trong các dự án Scratch.

Một vài ví dụ phổ biến khi sử dụng Hiển thị - Ẩn:

- Đối tượng chỉ hiển thị khi thoả mãn một điều kiện nào đó.



- Hiển thị một viên đạn cho tới khi nó kết thúc lộ trình và ẩn.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Hiển thị** và lệnh **Ẩn**.

d. Lệnh Đặt ra trước/ Go to front

Một vài ví dụ phổ biến sử dụng lệnh Đặt ra trước:

- Đưa một đối tượng hiển thị gần nhất với người xem trong một đoạn phim hoạt hình.



- Đưa một nút lệnh lên phía trước một thanh công cụ.



Lệnh này thường được sử dụng kết hợp cùng với lệnh **Ra sau () lớp** nhằm di chuyển đối tượng về phía sau các lớp xác định.

Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh Đặt ra trước.

e. Lệnh Ra sau () lớp/ Go Back () Layers

Một số ví dụ phổ biến sử dụng lệnh Ra sau () lớp:

- Làm cho một đối tượng nằm dưới các đối tượng khác.
- Khối này thường được kết hợp với lệnh **Đặt ra trước** trong các dự án Scratch.

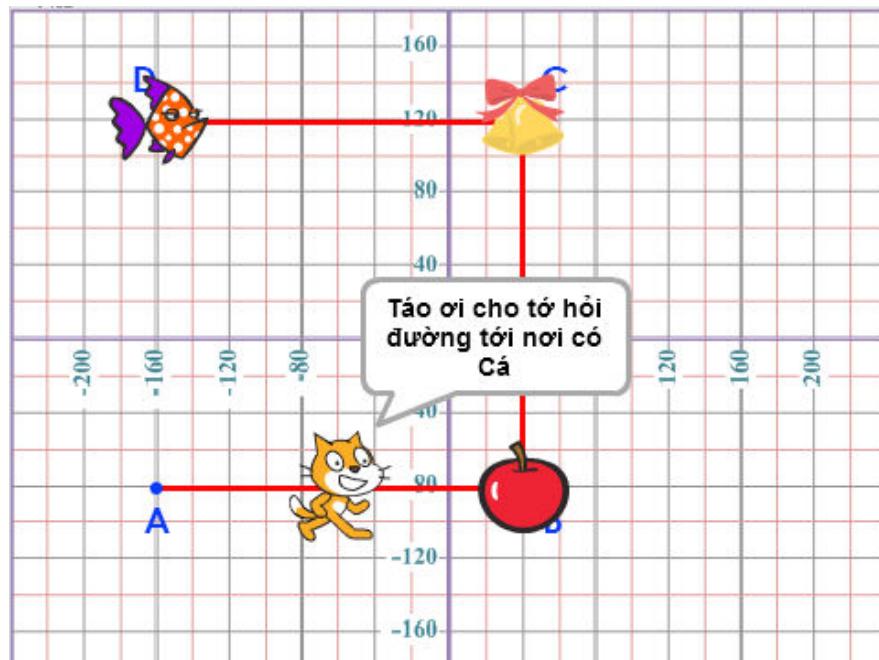


Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Ra sau () lớp**.

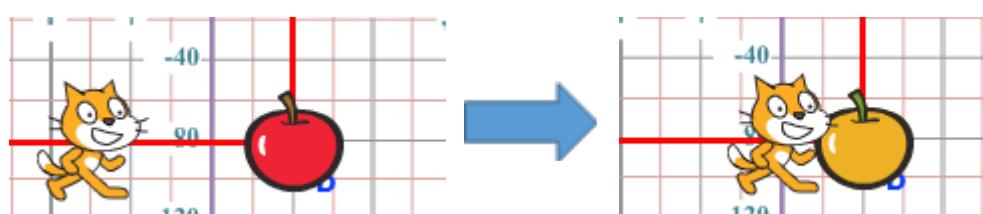
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Mở bài “Di chuyen 4” và thực hành sử dụng các lệnh đã học để trả lời các câu hỏi trong phần 2 nhỏ của “Hoạt động tìm tòi, khám phá”:

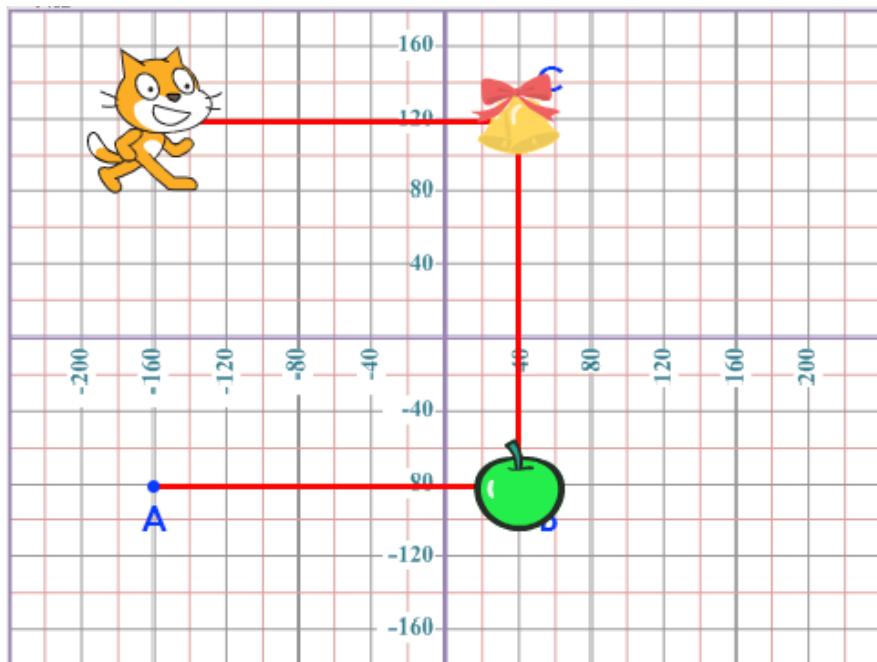
➤ Tạo sự giao tiếp giữa các đối tượng như một cuộc hội thoại thông thường (khi gặp các đối tượng, chú mèo đều phải chào hỏi và giao tiếp với đối tượng đó).



➤ Thay đổi màu sắc, kích thước của quả táo khi chạm vào con mèo



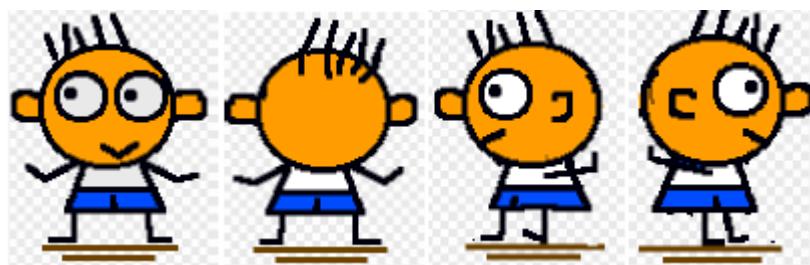
- Làm cho chuông vàng rung khi chú mèo chạm vào.
- Làm cho con mèo sẽ ăn con cá khi nó tìm thấy con cá.
- Làm cho con mèo lớn hơn sau khi ăn cá.



- Sau đó Chú mèo sẽ di chuyển tới vị trí ban đầu.
- Lưu lại bài với tên mới “Meo tim Ca”.

2. Thiết kế một chương trình nhảy theo nhạc:

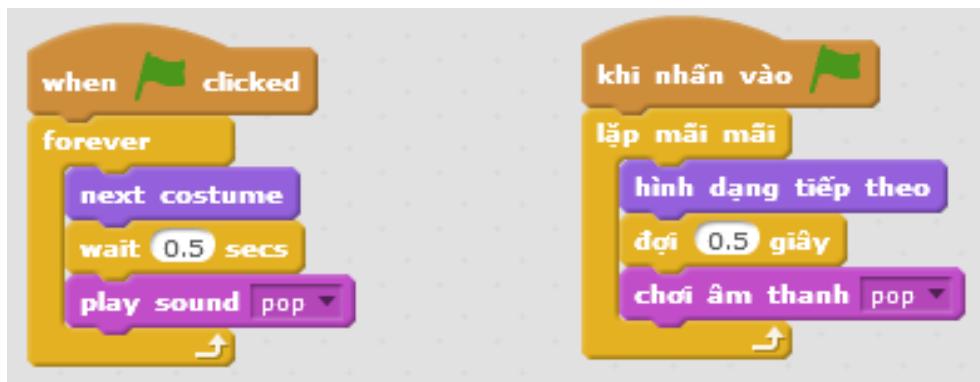
- Tạo đối tượng với các tư thế: trước, sau, trái, phải.



- Sử dụng các lệnh về ngoại hình để thay đổi tư thế.



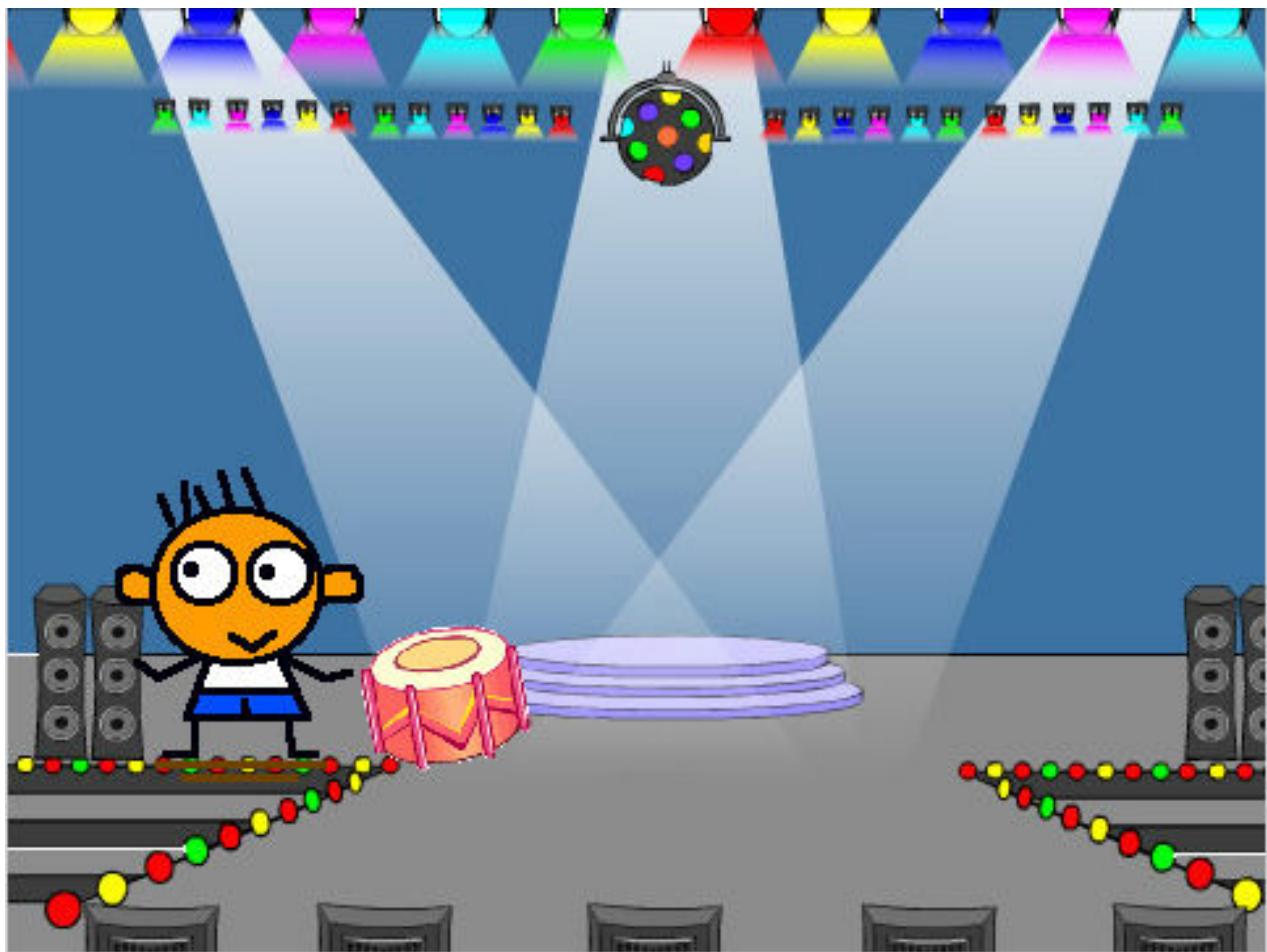
- Thủ thêm âm thanh vào cho đối tượng trong quá trình nhảy.



- Bây giờ hãy chạy thử chương trình, sau đó lưu lại chương trình với tên “**Nhay hip-hop**”.

3. Mở lại chương trình “Nhay hip-hop” và thêm hình nền sân khấu cho chương trình.

- Chọn **Spotlight-Stage** (sân khấu chiếu đèn) cho sân khấu



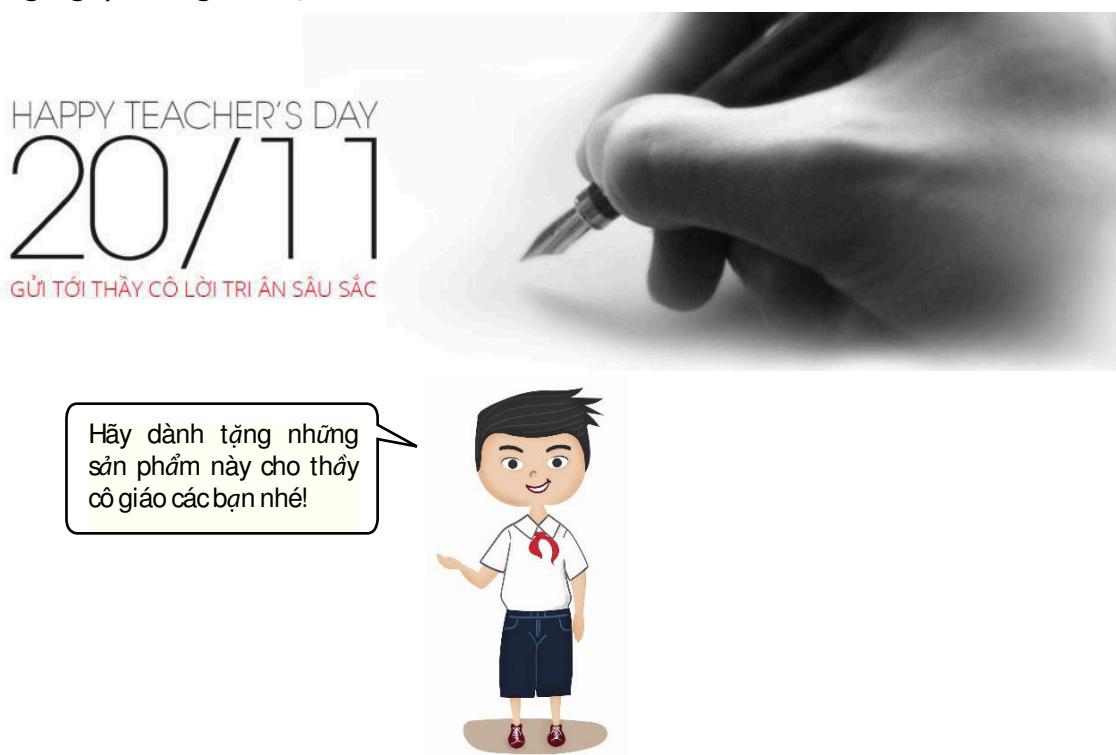
- Thay đổi hiệu ứng màu sắc cho sân khấu, giúp chương trình hấp dẫn hơn.
- Lưu lại chương trình với tên “**Nhay hip-hop 2**”.

C. HOẠT ĐỘNG SÁNG TẠO

Các bài tập ứng dụng – dự án hoạt động sáng tạo trong Mô đun này sẽ được thực hiện dưới dạng một dự án cá nhân. Mỗi học sinh sẽ tự lên ý tưởng, thiết kế và lập trình một trò chơi ứng dụng các kiến thức đã học ở Mô đun này và Mô đun trước.

1. Chủ đề

Ngày 20/11 hàng năm là ngày “Nhà giáo Việt Nam” nhằm thể hiện truyền thống “tôn sư trọng đạo”. Bằng việc sử dụng chương trình Scratch, bạn hãy xây dựng một chương trình, một trò chơi, một thiệp chúc mừng, hay một đoạn phim ngắn về chủ đề 20/11 để chúc mừng ngày Nhà giáo Việt Nam.



2. Yêu cầu bắt buộc

- Dự án phải bao gồm ít nhất hai đối tượng trở lên.
- Dự án phải có sự tương tác giữa các đối tượng, giữa người chơi và máy tính.
- Dự án áp dụng các kiến thức nắm được từ Mô đun 1 – Bài toán chuyển động và Mô đun 2 – Hội thoại và Hiệu ứng đồ họa.
- Sử dụng các hiệu ứng đồ họa như một dạng trang phục khác của đối tượng.

3. Lên ý tưởng và xây dựng kịch bản

- Tự lên ý tưởng cho chương trình xoay quanh chủ đề “Ngày nhà giáo Việt Nam”, có thể tham khảo ý kiến thầy cô và các bạn trong lớp.
- Xây dựng kịch bản, bao gồm: Bối cảnh, đối tượng, hoạt động, âm thanh nào.

- Phân tích các sự kiện, hành vi của đối tượng.
- Chọn và sử dụng âm thanh, hiệu ứng đồ họa cho chương trình.

4. Xây dựng chương trình

- Thực hiện chương trình bám sát kịch bản.
- Sử dụng các lệnh đã học trong Mô đun này và các Mô đun trước.
- Trình bày dự án trước lớp.

MÔ ĐUN 4. CHƠI NHẠC VÀ VẼ HÌNH TRONG SCRATCH

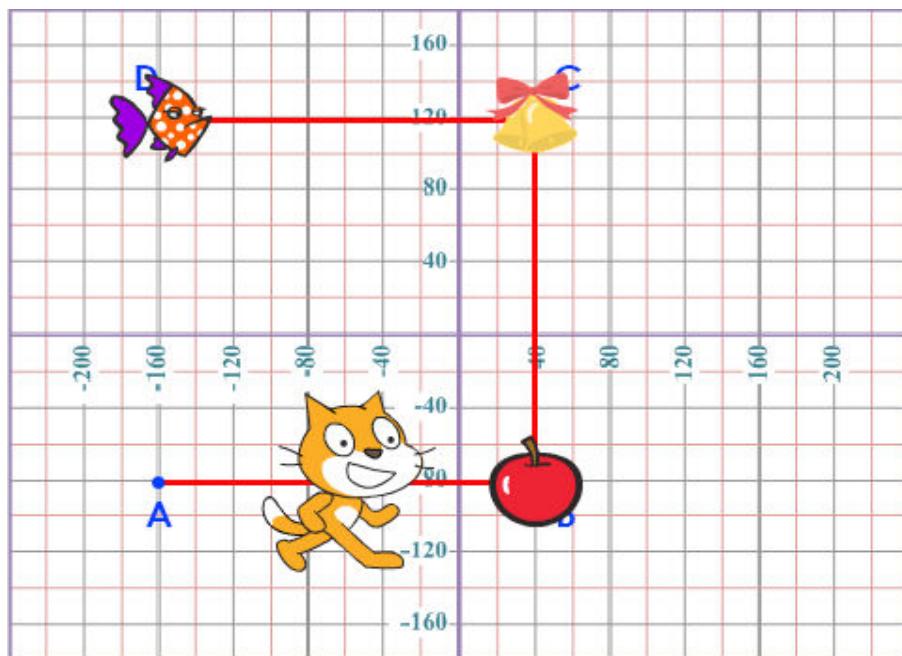
MỤC TIÊU

- Khám phá cách chơi nhạc và vẽ hình trong Scratch.
- Sử dụng các hiệu ứng âm thanh để làm tăng sức hút của các dự án.
- Vẽ và quản lý màu sắc trong Scratch.
- Thay đổi âm lượng khi chơi nhạc.
- Thiết kế được các chương trình chơi nhạc hoặc chương trình vẽ hình nghệ thuật.
- Thêm âm nhạc vào các dự án Scratch.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Quan sát và trả lời câu hỏi

Mở lại bài tập “Meo tim Ca” ở Mô đun trước và quan sát đường đi và các hoạt động của chú mèo.



- Lộ trình di chuyển của chú mèo có hình gì?
- Trên lộ trình, có những đoạn hội thoại, với những đối tượng nào?
- Chúng ta có thể lồng âm thanh vào các đoạn hội thoại đó không?
- Có thể vẽ lại lộ trình di chuyển của chú mèo không?

2. Tìm hiểu các nhóm lệnh

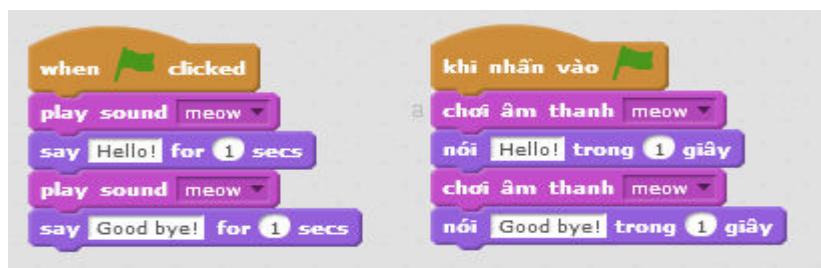
2.1. Chơi nhạc

Phần này sẽ giúp chúng ta tìm hiểu về các lệnh dùng để thêm nhạc, âm thanh vào dự án, đồng thời giúp chúng ta chỉnh sửa, thay đổi nhịp điệu của âm nhạc. Hãy quan sát ví dụ và thực hiện lại nó trên máy tính của bạn, sau đó đưa ra nhận định về chức năng, công dụng của từng lệnh ở cuối mỗi phần trình bày.

a. Lệnh Chơi âm thanh ()/ Play Sound ()

Sau đây là một số ví dụ phổ biến sử dụng lệnh Chơi âm thanh () .

- Chơi một đoạn hiệu ứng âm thanh tương ứng với một hành động của đối tượng.



- Tăng thêm mức độ của một sự kiện.



- Khi một đối tượng đạt được một điểm số hoặc mục tiêu thì một hiệu ứng âm thanh được phát ra.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chơi âm thanh ()**.

b. Lệnh chơi âm thanh () cho đến hết / Play Sound () Until Done

Một số ví dụ phổ biến sử dụng lệnh Chơi âm thanh () cho đến hết:

- Lặp lại một đoạn nhạc nền.



- Sử dụng lệnh này để chơi một đoạn nhạc nền để tạo độ trễ thay cho việc sử dụng lệnh **Chờ () giây** trong kịch bản.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chơi âm thanh () cho đến hết**.

c. Lệnh Ngừng mọi âm thanh/ Stop all Sounds

Một vài ví dụ phổ biến sử dụng Ngừng mọi âm thanh:

- Dừng chơi âm thanh trước khi chuyển sang hoạt cảnh tiếp theo (được sử dụng trong các đoạn phim hoạt hình).



- Dừng các bài hát đang được chơi trong một dự án về máy nghe nhạc.



- Trong các dự án cho phép bạn thiết lập các tùy chọn chơi nhạc hoặc tắt nhạc.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Ngừng mọi âm thanh**.

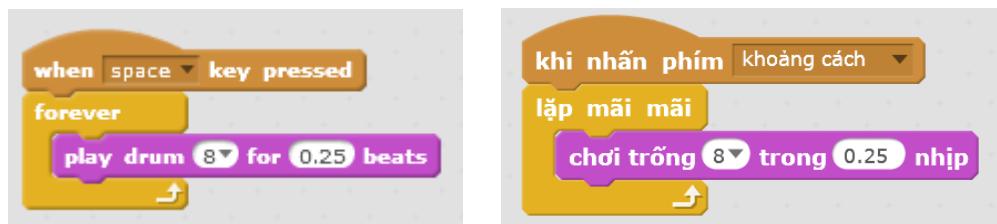
2.2. Chơi nhạc cụ

a. Lệnh Chơi trống () trong () nhịp / Play Drum () for () Beats

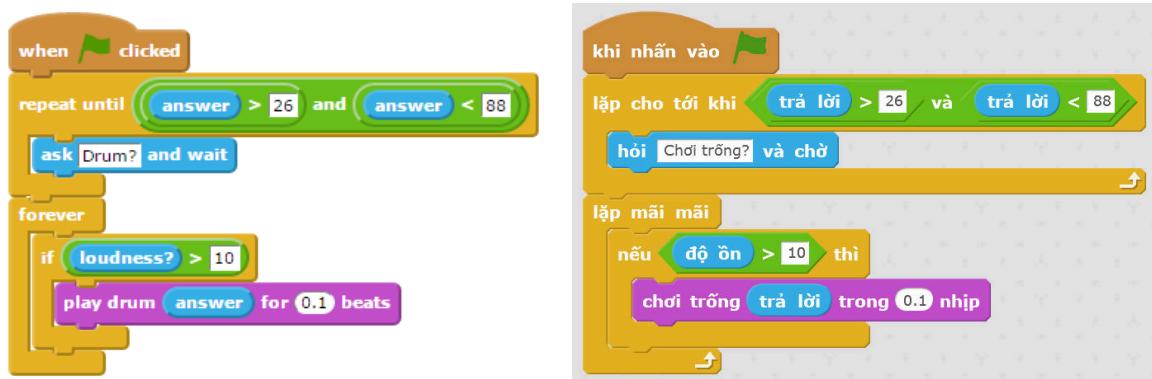
Lệnh này được gọi là chơi trống, tuy nhiên nó đã được thay đổi vì trường độ của nhịp điệu có thể bị thay đổi bởi lệnh **Đặt nhịp độ bằng () nhịp/phút**.

Một số ví dụ phổ biến sử dụng lệnh Chơi trống () trong () nhịp

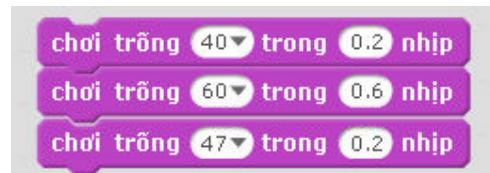
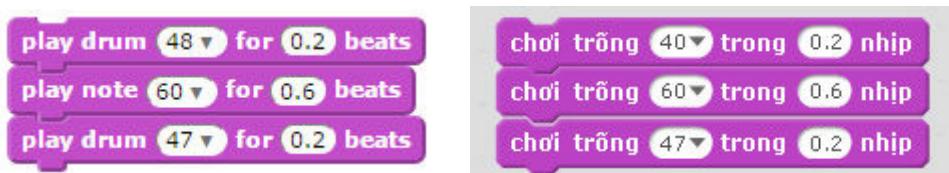
- Chơi một hiệu ứng âm thanh (ví dụ hiệu ứng vỗ tay – hiệu ứng số 8).



- Kiểm soát dự án âm nhạc.



- Tạo ra một bản nhạc được thực hiện bởi các lệnh.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chơi trống () trong () nhịp**.

b. Lệnh Chơi nốt nhạc () trong () nhịp/ Play Note () for () Beats

Một vài ví dụ phổ biến sử dụng lệnh này:

- Tạo những hiệu ứng âm thanh đơn giản, tạo các nốt cao thấp để tạo sự hấp dẫn, thú vị của âm nhạc.



- Các dự án âm nhạc mà bạn có thể điều khiển được nhạc cụ.

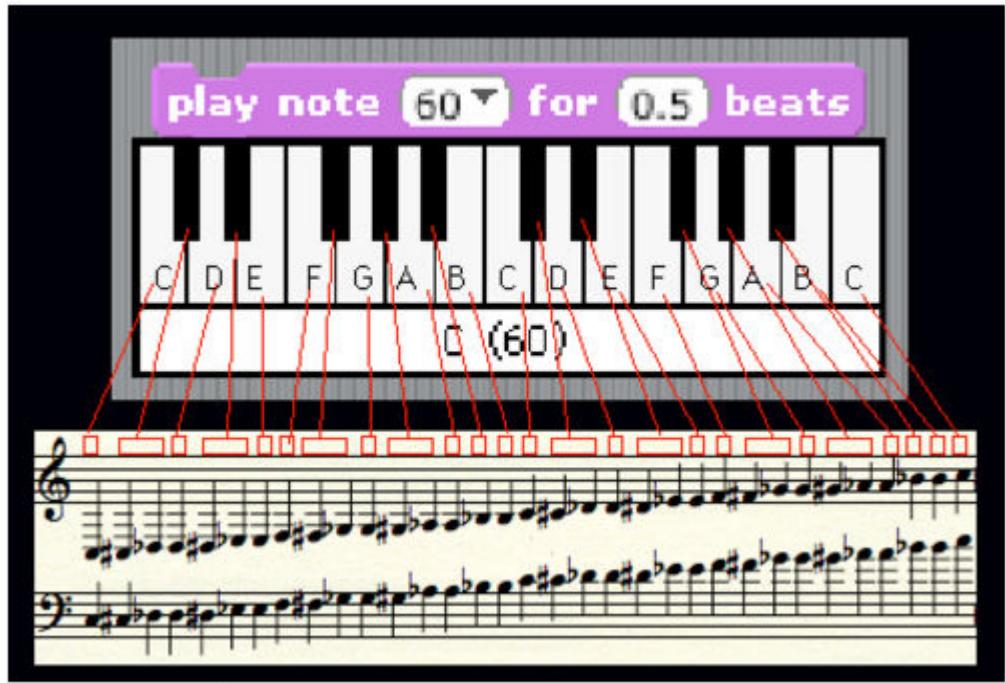


- Một bản nhạc được tạo ra bởi các lệnh.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chơi nốt nhạc () trong () nhịp**.

Lưu ý: **Chơi nốt nhạc () trong () nhịp** có thể sẽ tạo ra đôi chút khó khăn cho người sử dụng vì các ký hiệu và con số không giống như những bản nhạc thực tế.



c. Lệnh Đổi âm lượng ()/ Change Volume by ()

Một vài ví dụ điển hình khi sử dụng lệnh Đổi âm lượng ():

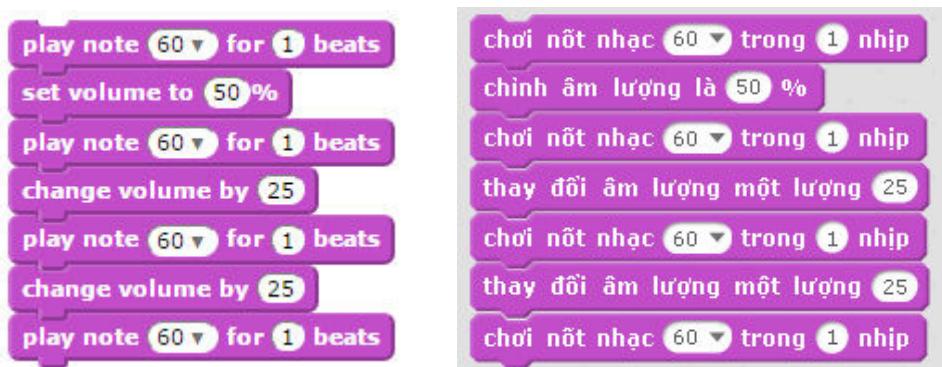
- Điều khiển âm lượng.



- Khi giảm kích thước và đồng thời giảm âm lượng của đối tượng.



- Tạo một đoạn nhạc bằng các lệnh.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh Đổi âm lượng () .

2.3. Vẽ hình và kiểm soát màu

Phần này giới thiệu các khối lập trình cho phép chúng ta sử dụng các bút vẽ, màu sắc và kiểm soát chúng bằng cách thiết lập hoặc thay đổi nét vẽ, màu sắc đồ họa... Quan sát các ví dụ và thực hiện lại chúng trên máy tính của bạn, sau đó đưa ra nhận định về các chức năng, công dụng của mỗi lệnh ở cuối mỗi phần trình bày.

a. Lệnh **Bắt đầu vẽ/ Pen down**

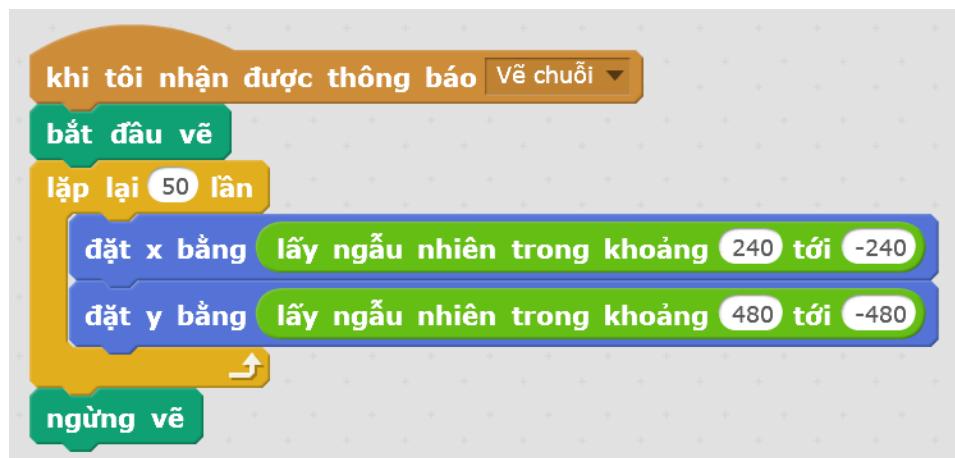
Lệnh **Bắt đầu vẽ** có tần suất sử dụng nhiều nhất trong các dự án vẽ nghệ thuật hoặc thiết kế đồ họa và cũng được dùng phổ biến trên một đối tượng, một đoạn kịch bản, hay một trang phục...

Một vài ví dụ phổ biến sử dụng lệnh Bắt đầu vẽ:

- Vẽ một chuỗi, đường kẻ (string) trong một hoạt cảnh hoạt hình.



- Vẽ hình trên màn hình, sân khấu.



- Vẽ các thanh trên đối tượng hoặc một đoạn kịch bản.



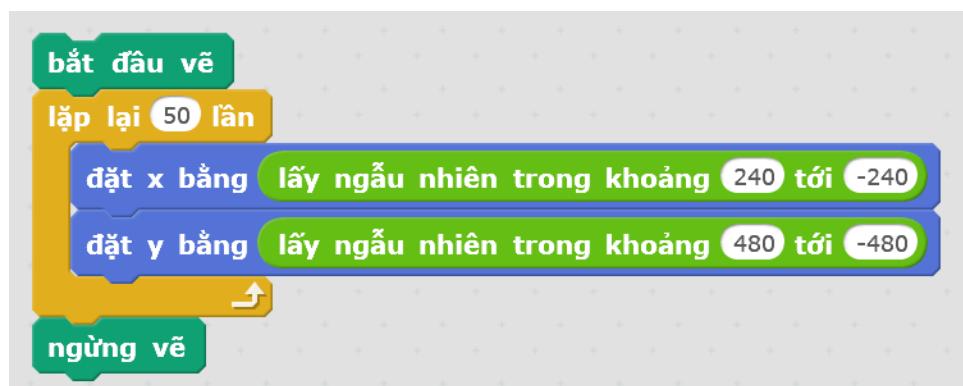
Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Bắt đầu vẽ**.

b. Lệnh Ngừng vẽ/ Pen up

Lệnh **Ngừng vẽ** không được dùng phổ biến trong các dự án Scratch, tuy nhiên nó vẫn có thể có nhiều tác dụng khi sử dụng các nhóm lệnh vẽ đồ họa... Nó thường được sử dụng cùng với lệnh **Bắt đầu vẽ**.

Ví dụ phổ biến sử dụng lệnh Ngừng vẽ:

- Kết thúc việc vẽ hình trên sân khấu.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Ngừng vẽ**.

c. Lệnh Chọn bút màu ()/ Set Pen Color to ()

Lệnh này được sử dụng chủ yếu khi các cây bút cần vẽ một đối tượng với màu sắc khác biệt mà chúng ta mong muốn.

Một vài ví dụ phổ biến sử dụng Chọn bút màu ():

- Chọn màu cho dự án vẽ.



- Tạo những đối tượng sử dụng những màu sắc khác biệt (ví dụ lửa và băng).



- Chọn màu cho một đối tượng được vẽ bằng bút (pen).



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chọn bút màu ()**.

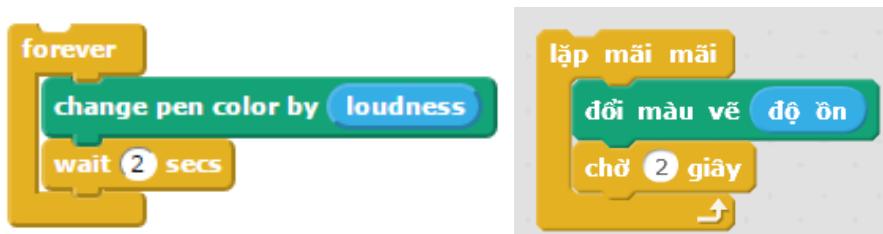
d. Lệnh Đổi màu vẽ ()/ Change pen color by ()

Những ví dụ phổ biến sử dụng lệnh Đổi màu vẽ ():

- Thay đổi màu sắc trong một khoảng thời gian nhất định để tạo ra những màu sắc nghệ thuật.



- Thay đổi màu sắc bút vẽ cho một chương trình nghệ thuật.



- Thay đổi màu sắc cho các đối tượng trong một dự án vẽ nghệ thuật với bút vẽ (Pen).



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi màu vẽ ()**.

2.4. Thay đổi kích thước, sắc thái nét vẽ và hiệu ứng in hình

Phần này giới thiệu các lệnh cho phép chúng ta thiết lập hoặc thay đổi sắc thái của cây bút. Giá trị của thuộc tính này nằm trong khoảng từ 0 (rất đậm hoặc tối đen) đến 100 (rất sáng). Ngoài ra chúng ta cũng có thể tìm hiểu về cách thiết lập hoặc thay đổi kích thước bút vẽ và làm thế nào để in hình đối tượng lên sân khấu. Hãy quan sát và thực hiện lại các ví dụ, sau đó đưa ra nhận định về chức năng, công dụng của các lệnh ở cuối mỗi phần trình bày.

a. Lệnh Đổi độ đậm màu vẽ một lượng ()/ Change pen shade by ()

Một vài ví dụ phổ biến thường dùng lệnh Đổi độ đậm màu vẽ một lượng ().

- Liên tục thay đổi màu bút khi chúng ta đang vẽ các bức tranh nghệ thuật với các sắc thái khác nhau.



- Thay đổi sắc thái của bút cho một chương trình nghệ thuật.



- Thay đổi sắc thái của đối tượng trong một dự án sử dụng bút vẽ.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đổi độ đậm màu vẽ một lượng ()**.

b. Lệnh Chính độ đậm màu vẽ thành () / Set pen Shade to ()

Một số ví dụ thường dùng lệnh Chính độ đậm màu vẽ thành ():

- Lặp lại việc chọn sắc thái bút vẽ khi sử dụng nó để tạo ra hình nghệ thuật với các sắc thái khác nhau.



- Thiết lập sắc thái cho đối tượng trong một dự án vẽ hình bằng bút vẽ (pen).



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chính độ đậm màu vẽ thành ()**.

c. Đổi kích cỡ nét vẽ - Change pen size by ()

Một số ví dụ phổ biến sử dụng lệnh Đổi kích cỡ nét vẽ ():

- Lặp lại việc thay đổi nét vẽ trong một dự án vẽ hình nghệ thuật.
- Thay đổi nét vẽ của đối tượng trong một dự án vẽ hình.

d. Hiệu ứng in hình – Stamp

Lệnh này được dùng để in hình đối tượng lên sân khấu. Khi được sử dụng trong một kịch bản, đối tượng sẽ tạo ra một hình ảnh của chính nó và được in lên trên Sân khấu. Các hình ảnh này không thể lập trình được vì nó không được coi là một đối tượng. Giống như các lệnh khác trong nhóm **Bút vẽ** không thể in hình lên một đối tượng khác. Các hình ảnh này có thể bị xoá đi bằng lệnh **Xóa**.

Một số ví dụ phổ biến

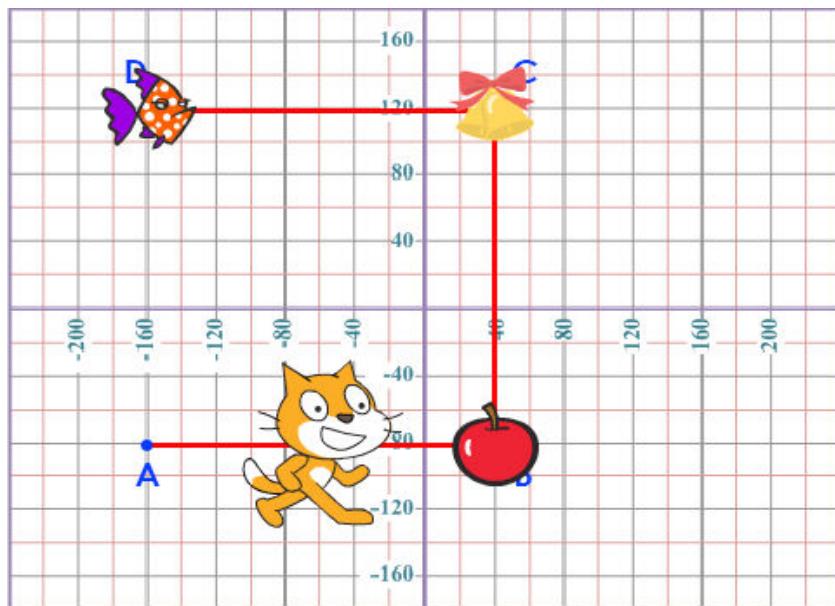
- Tạo ra nhiều hình ảnh của đối tượng trên màn hình.
- Tạo ra các hiệu ứng.
- Tạo hình trong các dự án vẽ hình.
- Tạo cảm giác chuyển động với nhiều đối tượng.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Mở lại bài “Meo tim Ca” và thực hiện lại theo yêu cầu sau:

- Vẽ lại lộ trình di chuyển của chú mèo.

- Lồng âm thanh vào các đoạn hội thoại của đối tượng.
- Thêm âm thanh khi chú mèo di chuyển (tiếng bước chân).



2. Lập trình chương trình “Chú mèo vẽ hình” theo yêu cầu.



Hình. Sử dụng các lệnh vẽ màu sắc và nét vẽ để vẽ hình

3. Mở lại bài “Nhay Hip-hop 2”.

- Thêm đối tượng tên là Hip-hop vào chương trình.
- Thêm âm nhạc vào cho đối tượng Hip-hop



- Lưu bài với tên “Nhay Hip-Hop3”.

C. HOẠT ĐỘNG SÁNG TẠO

1. Chủ đề

Hãy sử dụng phần mềm Scratch để xây dựng một đoạn phim hoặc một trò chơi để mô phỏng những hoạt động của đêm Giáng sinh.



Gợi ý cho đoạn phim: Vào dịp lễ "Giáng sinh", bạn nhận được một món quà, đó là một chiếc tàu hỏa đặc biệt. Chiếc tàu này có thể đưa bạn xuyên qua thời gian và không gian đến bất cứ nơi đâu trong vũ trụ, từ quá khứ đến hiện tại và cả tương lai. Bằng trí tưởng tượng của mình và thông qua phần mềm Scratch, bạn hãy làm một đoạn phim ngắn miêu tả những chuyến phiêu lưu kì thú của mình cùng các bạn trong kì "Giáng sinh" đáng nhớ này nhé!

Gợi ý cho Trò chơi: Giáng sinh đến, bên cạnh việc được nhận những món quà vô cùng dễ thương, các bạn còn được đi chơi và tham gia vào nhiều trò chơi hấp dẫn. Thông qua phần mềm Scratch, các bạn hãy thể hiện lại những trò chơi ấy bằng chính đôi bàn tay của mình nhé!

2. Yêu cầu

Đoạn phim:

- Độ dài tối thiểu 2 phút, tối đa 5 phút.
- Nội dung bộ phim phải bám sát vào chủ đề.
- Bộ phim phải có ít nhất 4 đối tượng và 4 chuyển cảnh khác nhau.
- Khuyến khích việc sử dụng các hiệu ứng di chuyển đối tượng, chuyển hình nền, tương tác, giao tiếp giữa các đối tượng trong phim, lồng tiếng, âm thanh sinh động.

Trò chơi:

- Trò chơi phải có nội dung liên quan đến chủ đề Giáng sinh.
- Trò chơi không được có nội dung bạo lực và cần phù hợp với lứa tuổi.
- Trò chơi phải có khởi đầu và kết thúc, khuyến khích trò chơi có nhiều cấp độ (level) cho người chơi. Nếu làm trò chơi không giới hạn cần có phần kiểm tra dừng trò chơi.

- Khuyến khích sử dụng hiệu ứng di chuyển, tương tác giữa các đối tượng và các hiệu ứng âm thanh.

3. Lên ý tưởng và xây dựng kịch bản

Ý tưởng và kịch bản được xây dựng dựa trên chủ đề đã nêu theo mẫu tại mục lục.

4. Trình bày dự án

Thực hiện dự án và trình bày trước lớp.

Mô đun 5. ĐIỀU KHIỂN VÀ CẤU TRÚC ĐIỀU KHIỂN

MỤC TIÊU

- Hiểu rõ chức năng của các lệnh trong nhóm Sự kiện và nhóm Điều khiển.
- Sử dụng tốt các cấu trúc lặp.
- Sử dụng tốt các cấu trúc rẽ nhánh.
- Chuyển các hoạt động đơn lẻ của các đối tượng được lập trình trong các mô đun trước thành các vòng lặp (nếu phù hợp).
- Có thể hoàn thiện tốt các trò chơi hay sản phẩm học tập liên quan tới chuyển động.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Thủ nghiệm các lệnh

- Mở lại các bài toán vẽ hình ở Mô đun trước.
- Thủ thay đổi những hoạt động, kịch bản có tính quy luật, được lặp đi lặp lại nhiều lần bằng các vòng lặp trong nhóm lệnh Điều khiển (Control).
- Chạy thử và quan sát lại hoạt động của đối tượng, sau đó mô tả hoạt động đó bằng lưu đồ khối.

2. Tìm hiểu các nhóm lệnh

2.1. Nhóm lệnh sự kiện

Phần này giúp chúng ta tìm hiểu các lệnh dùng để điều khiển kịch bản và đối tượng. Hãy quan sát và thực hiện lại các ví dụ, sau đó đưa ra nhận định của bạn về chức năng, công dụng của các lệnh tương ứng ở cuối mỗi phần trình bày.

a. Lệnh khi nhấn vào lá cờ xanh / When green Flag Clicked

Khi dự án của bạn được bắt đầu, rất nhiều điều sẽ xảy ra, và những diễn biến xảy ra tại thời điểm này đều được kích hoạt bởi lệnh **Khi nhấn vào lá cờ xanh**.

Một số ví dụ phổ biến (Bạn hãy tự thực hiện ví dụ theo gợi ý)

- Bắt đầu chạy một dự án Scratch
- Thiết lập lại giá trị của các biến
- Xoá các hình vẽ được tạo từ bút vẽ

- Chạy một bản nhạc...

Lệnh này được sử dụng trong hầu hết tất cả các dự án Scratch.

Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Khi nhấn vào lá cờ xanh**.

b. Lệnh Khi nhấn phím ()/ When () Key pressed

Một vài ví dụ phổ biến sử dụng lệnh Khi nhấn phím ():

- Điều khiển hoạt động của một đối tượng.



- Xử lý văn bản.



- Bắt đầu một đoạn hoạt hình.



- Điều khiển di chuyển.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Khi nhấn phím ()**.

c. Lệnh Khi đối tượng này được nhấp vào / When This Sprite Clicked

Một vài ví dụ thường sử dụng lệnh Khi đối tượng này được nhấp vào:

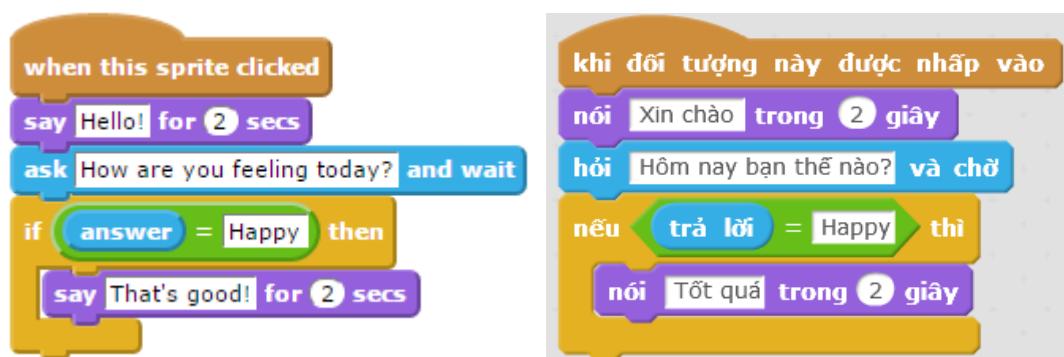
- Tạo một nút lệnh (ví dụ: tạo một thực đơn).



- Chuyển đổi sân khấu khi nháy chuột vào một nút lệnh.



- Điều khiển đầu vào.



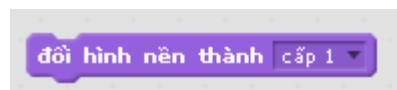
Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Khi đối tượng này được nhấp vào**.

d. Lệnh Khi hình nền đổi thành () / When Backdrop Switches to ()

Lệnh này nằm trong nhóm lệnh **Các trường hợp** (Events). Những đoạn mã gắn bên dưới lệnh này sẽ được kích hoạt khi sân khấu chuyển tới một hình nền xác định.

Một vài ví dụ phổ biến sử dụng lệnh Khi hình nền đổi thành ():

- Bắt đầu một cấp bậc trong một dự án trò chơi.
- Lập trình đổi với sân khấu:

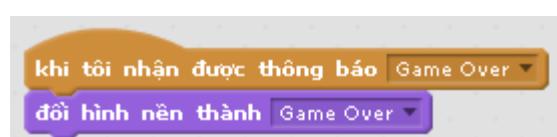


- Lập trình đổi với đối tượng :



- Khi kết thúc một trò chơi.

- Lập trình đổi với sân khấu:



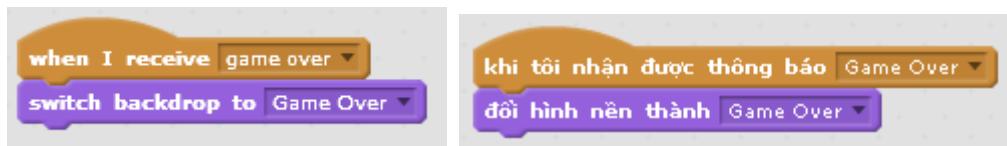
- Lập trình đối với đối tượng:



e. Lệnh Khi tôi nhận được thông báo ()/ When I receive ()

Một vài ví dụ phổ biến sử dụng lệnh Khi tôi nhận được thông báo ():

- Thực hiện việc chuyển đổi hình nền khi nhận được thông báo **Game Over**.



- Bắt đầu một hoạt động của đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Khi tôi nhận được thông báo ()**.

2.2. Nhóm cấu trúc điều khiển

Trong phần này, chúng ta sẽ được tìm hiểu về các cấu trúc lặp và cấu trúc rẽ nhánh, giúp chúng ta có nhiều giải pháp hơn trong việc xây dựng các hoạt động của đối tượng. Ngoài ra, phần này cũng giới thiệu cách thức để chúng ta có thể tạo ra các bản sao của đối tượng và sử dụng chúng.

Các bạn hãy quan sát ví dụ và cho biết chức năng của mỗi lệnh nhé!



a. Lệnh Lặp lại () lần/ Repeat ()

Một số ví dụ thường sử dụng lệnh Lặp lại () lần:

- Lặp lại một đoạn mã bất kỳ với số lần xác định.



- Liên tục kiểm tra di chuyển của đối tượng.



- Sử dụng trong các kịch bản vẽ hình.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Lặp lại () lần**.

b. Lệnh Lặp mãi mãi/ Forever

Lệnh này là một trong những lệnh được sử dụng nhiều nhất trong các dự án Scratch bởi vì có rất nhiều trường hợp cần sử dụng đến vòng lặp vô hạn này.

Một vài ví dụ phổ biến sử dụng lệnh Lặp mãi mãi

- Lưu điểm cao nhất trong một trò chơi.



- Dùng kiểm tra xem đối tượng đã chạm mặt đất (màu xanh) chưa?



- Lặp một bản nhạc.



- Lặp lại hoạt động đơn giản của đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Lặp mãi mãi**.

c. Lệnh Lặp cho tới khi ()/ Repeat Until ()

Một số ví dụ điển hình sử dụng lệnh Lặp cho tới khi ():

- Di chuyển nhân vật cho tới khi X hoặc Y đạt đến một giá trị nhất định.



- Nhảy tới vị trí con trỏ chuột cho tới khi nó không được nhấp.



- Lặp lại một đoạn mã trong một khoảng thời gian nhất định.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Lặp cho tới khi ()**.

d. Lệnh Nếu () thì / If () then

Một số ví dụ thường sử dụng lệnh Nếu () thì:

- So sánh các giá trị.



- Kiểm tra đầu vào.



- Điều khiển đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nếu () thì**.

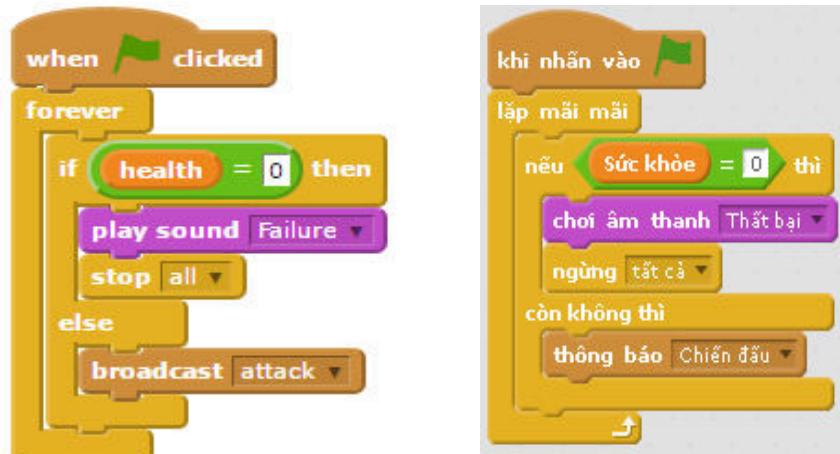
e. Lệnh Nếu () thì, còn không thì / If () then, Else

Sau đây là một vài ví dụ phổ biến sử dụng lệnh Nếu () thì, còn không thì:

- Thực hiện một việc khi thoả mãn một điều kiện xác định.



- Nếu sức khoẻ của một đối tượng bằng một giá trị nhất định, nó sẽ bị chết, ngược lại nó sẽ thực hiện công việc khác.



- Kiểm tra giá trị của một biến.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Nếu () thì, còn không thì**.

f. Lệnh Tạo bản sao từ () / Create Clone of ()

Một số ví dụ phổ biến sử dụng lệnh Tạo bản sao từ ():

- Không giới hạn số lượng viên đạn trong một trò chơi.



- Tạo nhiều bản sao của một đối tượng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Tạo bản sao từ()**.

g. Lệnh Khi tôi khởi đầu là một bản sao/ When I Start as a Clone

Một số ví dụ phổ biến sử dụng lệnh Khi tôi khởi đầu là một bản sao

- Đưa bản sao tới một vị trí ngẫu nhiên.



Trong ví dụ này, khi bản sao được tạo ra, nó sẽ được thiết lập vị trí ngẫu nhiên trên Sân khấu.

- Tạo một bản sao di chuyển cho tới khi chạm vào một đối tượng khác:



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Khi tôi khởi đầu là một bản sao.**

h. Lệnh xóa bản sao này /Delete this Clone

Một vài trường hợp phổ biến sử dụng lệnh Xóa bản sao này:

- Xoá đi một viên đạn khi nó bắn xong.



- Loại bỏ một đối tượng được tìm thấy trong một trò chơi trốn tìm.



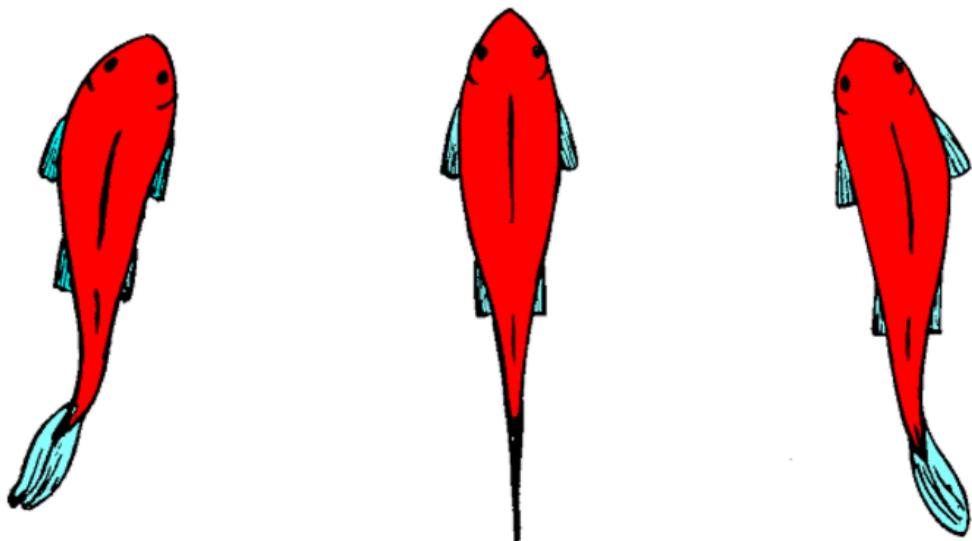
Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Xóa bản sao này**.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

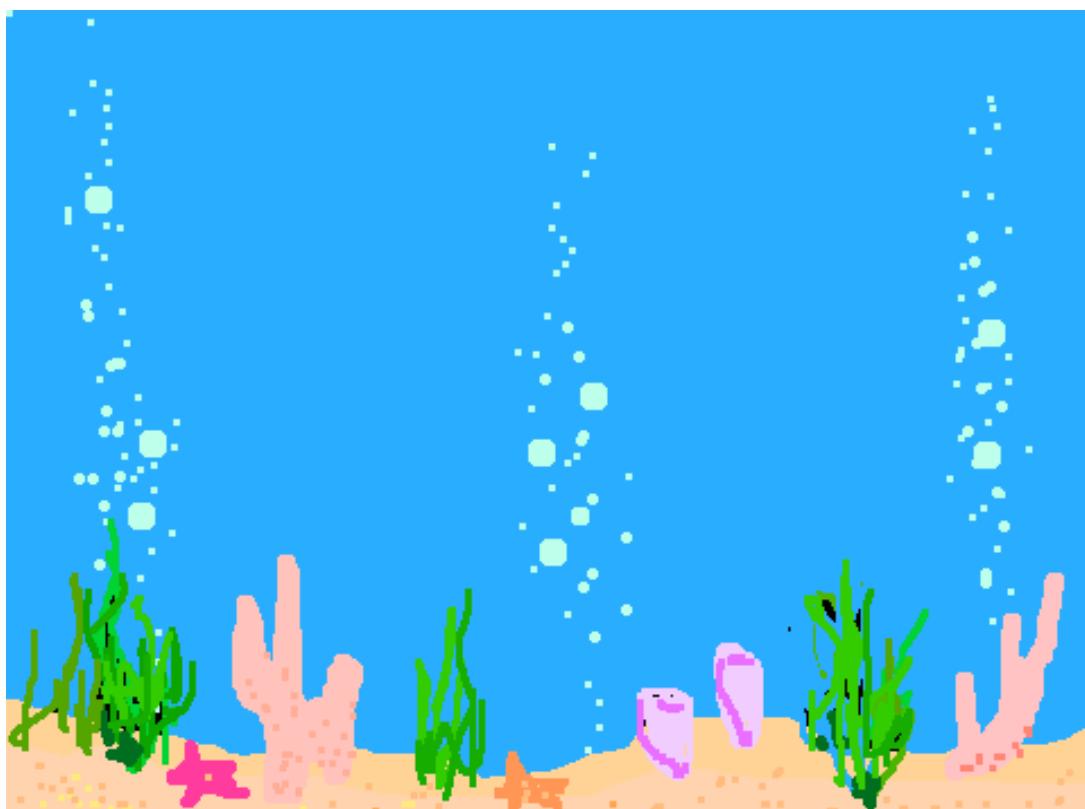
Thiết kế trò chơi “Bể nuôi cá cảnh”

Bạn thử hình dung một bể cá cảnh trên màn hình máy tính, trong đó những con cá bơi tung tăng. Hãy sử dụng Scratch để tạo ra một bể cá với nhiều màu sắc khác nhau để ngắm thoả thích.

- Trước tiên hãy vẽ Cá với 3 hình dạng khác nhau như hình minh họa.



➤ Thêm hình nền cho sân khấu, chúng ta nên chọn những hình nền dưới nước phù hợp với trò chơi.

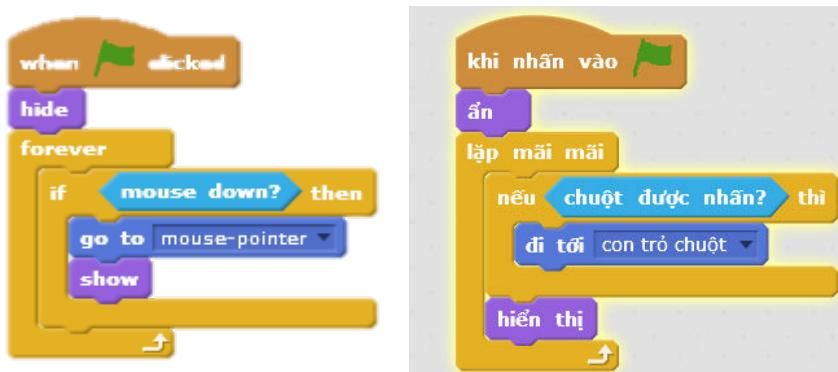


➤ Vẽ thêm thức ăn cho cá (có thể vẽ một hình tròn như mẩu bánh hoặc, các hạt thức ăn nhỏ).

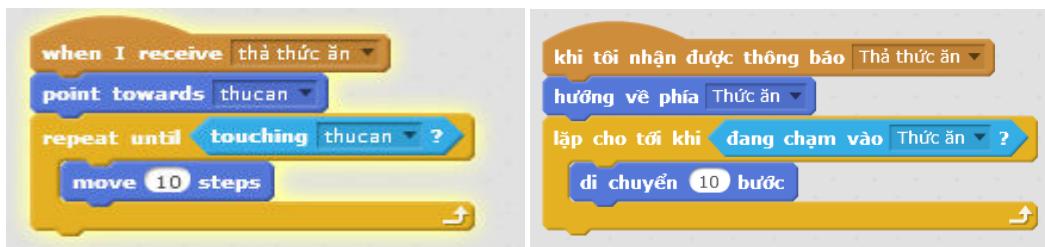
➤ Lập trình chuyển động cho cá theo gợi ý: Khi bơi, cá liên tục thay đổi ngoại hình (*giống như quẩy đuôi hoặc vây...*), chúng ta sẽ dùng lệnh **Hình dạng tiếp theo (Next Costume)** để thực hiện việc này. Trong quá trình di chuyển liên tục, sẽ có lúc cá chạm vào thành bể (*biên sân khấu*), cá sẽ xoay một góc xác định để chuyển hướng và di chuyển tiếp trong bể.



➤ Lập trình cho cá ăn: Bạn hãy hình dung cách cho cá ăn như sau: bạn dùng chuột bấm vào một vị trí bất kỳ trong "bể cá". Tại chỗ được bấm xuất hiện một mẩu "thức ăn" khiến cá vội vàng lao tới thức ăn. Điều này có nghĩa là thức ăn sẽ ẩn đi cho tới khi chúng ta bấm chuột vào bể cá thì mới xuất hiện.



➤ Lập trình cho hoạt động ăn thức ăn của cá: Khi thức ăn được thả, lũ cá lập tức lao đến chỗ thức ăn (dùng lệnh **Hướng về phía ()**)



- Lập trình cho thức ăn ẩn đi khi cá chạm vào chúng.
- Sau khi thực hiện xong, lưu bài với tên “Be nuoi Ca Canh”.

C. HOẠT ĐỘNG SÁNG TẠO

Chia lớp học thành các nhóm khác nhau – Mỗi nhóm gồm 5 thành viên (bao gồm 1 trưởng nhóm) thực hiện việc xây dựng ý tưởng sáng tạo về một dự án trò chơi hoặc một đoạn phim hoạt hình ý nghĩa, phù hợp với lứa tuổi.

1. Chủ đề

Bảo vệ môi trường là một đề tài nóng và rất cấp thiết hiện nay. Dựa trên phần mềm Scratch, các bạn hãy xây dựng một trò chơi hoặc một đoạn video về việc cải tạo và bảo vệ môi trường.

2. Yêu cầu

➤ Nội dung trò chơi và video khuyến khích sự sáng tạo, vui vẻ và mang tính nhân văn.

➤ Khuyến khích những trò chơi, video có cốt truyện và hướng dẫn cụ thể, rõ ràng.

Đối với đoạn Video

- Độ dài tối thiểu 1 phút rưỡi, tối đa 5 phút.

- Khuyến khích việc sử dụng các hiệu ứng di chuyển đổi tượng, chuyển hình nền, tương tác, giao tiếp giữa các đối tượng trong phim.

Đối với game:

- Khuyến khích sử dụng tương tác với người chơi theo nhiều cách khác nhau: chuột, bàn phím, âm thanh, camera....

- Khuyến khích sử dụng hiệu ứng di chuyển, tương tác giữa các đối tượng và các hiệu ứng âm thanh.

3. Xây dựng kịch bản

Ý tưởng và kịch bản phải được xây dựng trên cơ sở chủ đề “Bảo vệ môi trường” theo mẫu tại phụ lục.

4. Trình bày dự án

Thực hiện dự án và trình bày dự án trước lớp.

MÔ ĐUN 6. TOÁN HỌC TRONG SCRATCH

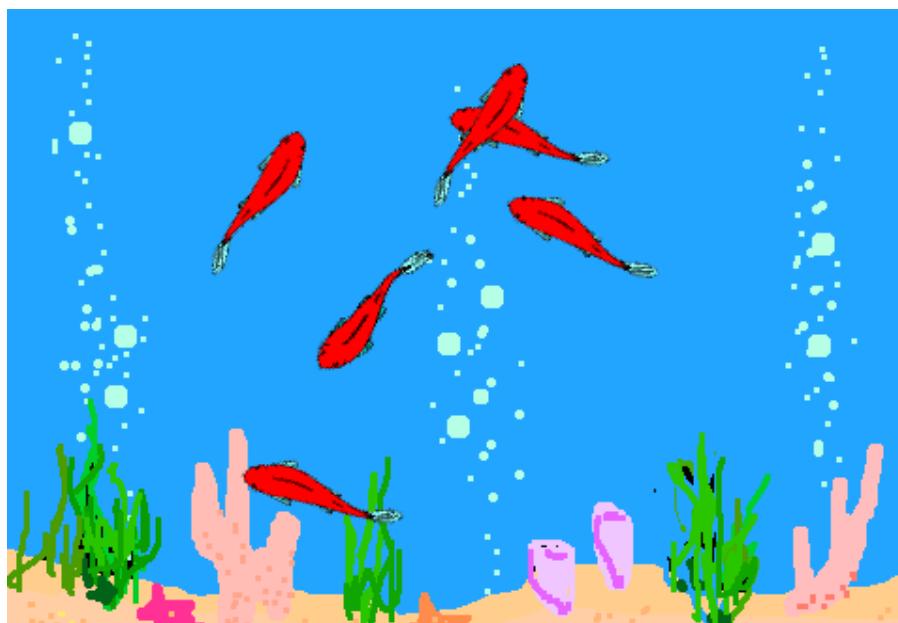
MỤC TIÊU

- Hiểu rõ chức năng các phép toán số học và logic trong nhóm lệnh toán học.
- Sử dụng tốt các lệnh toán học trong việc thực hiện các thao tác tính toán.
- Kết hợp với các cấu trúc điều khiển để tạo ra những dự án hoàn thiện.
- So sánh các giá trị của biểu thức toán học.
- Áp dụng toán học vào lập trình và sử dụng lập trình để học toán.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Quan sát

➤ Mở lại trò chơi “Nuôi Cá Cảnh” và quan sát hoạt động của những con cá trong bể.



- Sự di chuyển của các con cá trong bể diễn ra như thế nào? Có giống nhau không?
- Khi chạm thành bể cá con cá có quay theo một hướng nhất định không?
- Hãy sử dụng phép toán lấy ngẫu nhiên lấy ngẫu nhiên trong khoảng 1 đến 10 thay vào số bước của các con cá và quan sát.
- Hãy sử dụng phép toán lấy ngẫu nhiên lấy ngẫu nhiên trong khoảng 1 đến 10 thay vào góc quay của các con cá và quan sát.

2. Tính điểm

Hãy mở lại trò chơi **Flappy Bird** ở Mô đun trước, thử tìm cách tính điểm cho người chơi, và tìm cách lưu lại điểm cao nhất trong các lần chơi? Hãy mô tả bằng ngôn ngữ thông thường của bạn về cách tính điểm mà bạn đang định áp dụng cho trò chơi này.

3. Tìm hiểu các nhóm lệnh

3.1. Nhóm lệnh toán học

Các phép toán trong Scratch không chỉ giúp chúng ta thực hiện các hoạt động tính toán mà nó còn có vai trò rất quan trọng trong hầu hết các dự án Scratch. Các phép toán được chia thành hai loại: Kiểu logic (Boolean) và kiểu báo cáo (Reporter). Hãy quan sát và thực hiện lại các ví dụ để hiểu rõ hơn về chức năng, công dụng của từng loại.

a. Kiểu logic (Boolean)

Trong Scratch, các lệnh thuộc kiểu logic là những khối có hình lục giác, có chứa một điều kiện nào đó. Các lệnh này sẽ trả về giá trị đúng (true) hoặc sai (false) khi nó được gọi đến.

Trong Nhóm lệnh toán học có những lệnh kiểu logic sau:

	So sánh nhỏ hơn
	So sánh bằng
	So sánh lớn hơn
	Phép toán “và” (and): trả về giá trị Đúng khi cả 2 điều kiện là đúng, và trả về giá trị Sai nếu một trong 2 điều kiện là sai
	Phép toán “hoặc” (or): trả về giá trị Đúng khi 1 trong 2 điều kiện là đúng, và trả về giá trị Sai nếu cả 2 điều kiện là sai
	Phép toán phủ định: Trả về giá trị Đúng khi điều kiện sai và ngược lại

Một số ví dụ phổ biến:

- Lưu điểm cao nhất trong một trò chơi.



- Kiểm tra xem đối tượng đã chạm mặt đất (màu xanh) chưa.



b. Kiểu báo cáo (Reporter)

Trong Scratch, các khối dữ liệu thuộc kiểu báo cáo có dạng hình bầu dục, có chứa một phép toán nào đó. Lệnh này sẽ trả về một kết quả cụ thể với kiểu dữ liệu xác định như kiểu số, kiểu văn bản.

Trong nhóm lệnh toán học có những lệnh kiểu báo cáo như sau:

	Phép cộng: Trả về giá trị là tổng 2 số
	Phép trừ: Trả về giá trị là hiệu 2 số
	Phép nhân: Trả về giá trị là tích 2 số
	Phép chia: Trả về giá trị là thương 2 số
lấy ngẫu nhiên	Phép lấy ngẫu nhiên trong một khoảng
nối [hello và world]	Phép toán nối chuỗi
chữ cái thứ 1 của world	Phép toán trích ra ký tự thứ n của chuỗi
độ dài của world	Trả về độ dài của một chuỗi
	Phép chia lấy phần dư
làm tròn []	Phép làm tròn số
căn [] của []	Phép căn bậc 2 của một số

Một số ví dụ phổ biến sử dụng khối các lệnh báo cáo:

- Một đoạn mã tính toán.



- Công thức toán học.



- Kiểm tra xem số a có chia hết cho số b không.



- Kiểm tra một số a là chẵn hay lẻ.



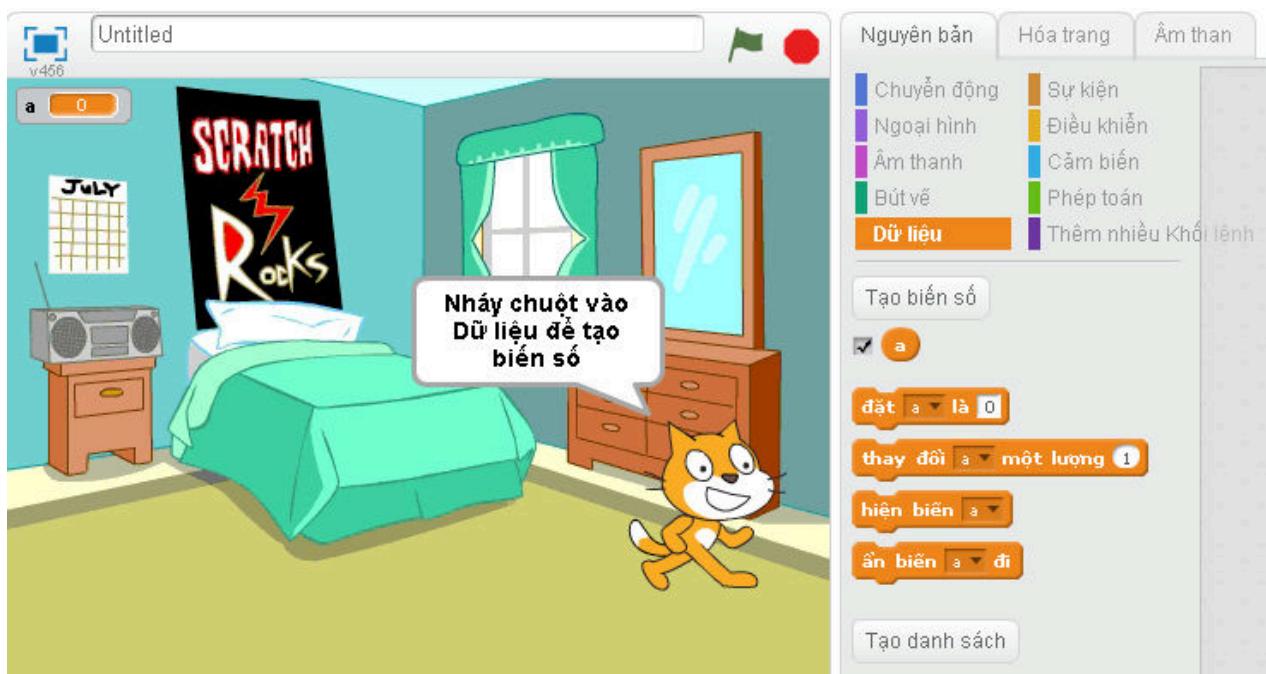
3.2. Sử dụng biến trong Scratch

Biến là một giá trị thay đổi được lưu trong bộ nhớ của Scratch. Biến chỉ có thể giữ một giá trị trong một thời gian nhất định. Nhấp vào tên biến trong khu vực kịch bản, chúng ta sẽ thấy hiện lên giá trị hiện tại của biến. Không giống như một số ngôn ngữ lập trình khác, các biến trong Scratch chỉ thực sự được khởi tạo khi dự án thực sự chạy. Điều này làm cho bộ nhớ RAM của máy tính không bị chiếm dụng nhiều bởi các giá trị của biến được lưu trữ cho dự án.

Các giá trị của biến có thể là số hoặc chuỗi văn bản bất kỳ.



Các biến được tạo ra khi chúng ta bấm chọn **Tạo biến số (Make a Variable)** trong khối **Dữ liệu (Data)**.



a. Lệnh Đặt () là () /Set () to ()



Bạn hãy nháy chuột vào mũi tên màu đen để chọn biến số nhé!



Một vài ví dụ phổ biến sử dụng lệnh Đặt () là ():

- Thiết lập lại các giá trị của dự án.



- Chọn cấp bậc trong một trò chơi.



- Thiết lập giá trị cho một công thức toán học.

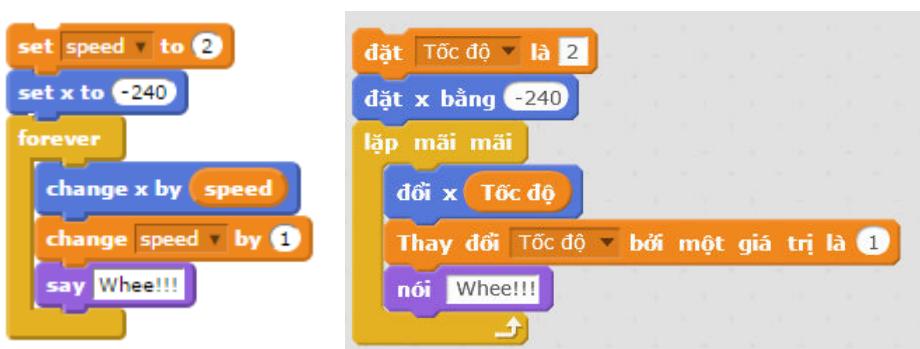


Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đặt () là ()**.

b. Lệnh Thay đổi () bởi một giá trị là ()/ Change () by ()

Một vài ví dụ phổ biến sử dụng lệnh Thay đổi () bởi một giá trị là ():

- Thay đổi tốc độ của một đối tượng.



- Thay đổi cấp bậc của một trò chơi.



- Thay đổi điểm số trong một trò chơi.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Thay đổi () bởi một giá trị là ()**.

c. Lệnh Hiện biến ()/ Show Variable (); Ẩn biến () đi/ Hide Variable ()

Một vài ví dụ phổ biến sử dụng lệnh Hiện biến () và Ẩn biến () đi:

- Hiển thị điểm số khi trò chơi kết thúc.



- Ẩn các biến khi một người chơi kết thúc game.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Hiện biến ()** và **Ẩn biến () đi**.

3.3. Sử dụng danh sách trong Scratch

Một danh sách (một số ngôn ngữ lập trình khác gọi là *mảng*) là một kiểu dữ liệu hay một công cụ có thể được sử dụng để lưu trữ nhiều thông tin cùng lúc. Một danh sách bao gồm nhiều phần tử khác nhau. Mỗi một phần tử có thể truy xuất thông qua chỉ số của phần tử đó. Giá trị của các phần tử này có thể nhìn thấy trên bảng dữ liệu của danh sách.

Ví dụ D: Mảng Nhietdo có 6 phần tử từ 1 đến 6 với các giá trị khác nhau:

Nhietdo	
1	25
2	30
3	27
4	28
5	34
6	40
+ length: 6	

a. Các lệnh của danh sách

Trong kiểu dữ liệu danh sách bao gồm các lệnh như sau:

	Thêm một phần tử vào danh sách
	Xoá đi phần tử thứ n trong danh sách
	Chèn một phần tử vào vị trí thứ n của danh sách
	Thay thế phần tử thứ n bởi một phần tử khác
	Trích xuất phần tử thứ n của danh sách
	Trả về độ dài của danh sách
	Kiểm tra trong danh sách có bao gồm phần tử...
	Hiện danh sách
	Ẩn danh sách

b. Các phần tử trong danh sách

Các phần tử trong danh sách có thể thêm vào hoặc xoá đi, bằng tay hoặc bằng cách lập trình. Nháy chuột vào một phần tử của danh sách sau đó nhấn giữ **Shift** và nhấn **Enter** chúng ta sẽ thêm được các phần tử vào trước phần tử được lựa chọn.

Bạn cũng có thể thêm các phần tử bằng cách nháy phải chuột vào danh sách và chọn **Import** – có thể đưa vào một tệp định dạng **.txt** hoặc **.csv**. Mỗi một dòng trên tệp sẽ là một phần tử của danh sách. Bạn cũng có thể xuất ra (**Export**) danh sách này thành một tệp bằng cách tương tự.

c. Giới hạn kích thước trên danh sách

Danh sách trong Scratch được coi là không có giới hạn về độ dài hay số lượng các phần tử có thể lưu trữ trong một danh sách. Tuy nhiên, bạn sẽ không thể lưu một dự án trực tuyến nếu danh sách trong dự án quá dài để đăng tải lên. Khi đó, Scratch sẽ thông báo lỗi kết nối mạng sau khoảng 30 giây.

Các ví dụ thường dùng

- Tính toán.
- Mã hoá/ giải mã một chuỗi.
- Danh sách các mặt hàng.

- Trí tuệ nhân tạo.
- Ghi nhiều giá trị.
- Bảo tồn giá trị của các biến.
- Lưu vào tệp dữ liệu.

3.4. Nhóm lệnh cảm biến

a. Lệnh Đang chạm vào màu ()? / Touching Color ()?

Lệnh này được sử dụng rộng rãi trong các dự án Scratch đặc biệt là các dự án dạng trò chơi cần đối tượng vượt qua các chướng ngại vật trong địa hình (platformer).

Một vài ví dụ phổ biến dùng Đang chạm vào màu ()?:

- Di chuyển một nhân vật cho tới khi nó chạm vào một màu xác định.



- Đối tượng sẽ làm một công việc nào đó khi chạm vào màu xác định



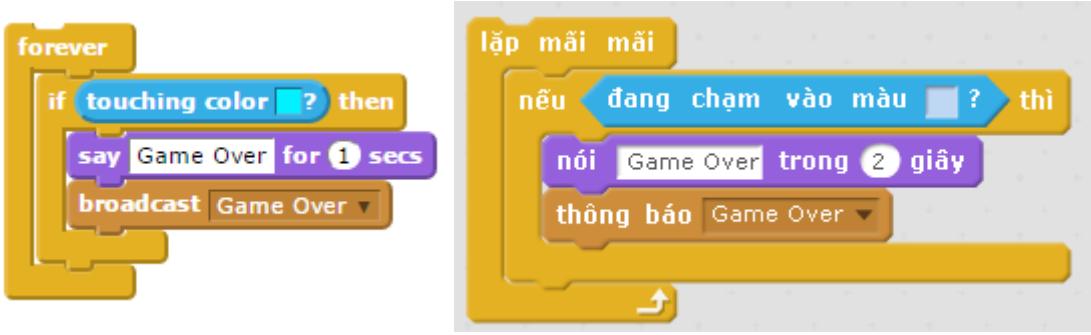
- Dừng bắn đạn khi nó chạm vào một bức tường màu đỏ.



- Thông báo "Ngõ cụt" khi đi trong một mê cung.



- Thông báo **Game Over** khi đối tượng chạm vào nước (màu xanh).



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Đang chạm vào màu ()?**.

b. Lệnh Màu () chạm vào màu ()? / Color () is Touching ()?

Một vài ví dụ phổ biến sử dụng lệnh này.

- Di chuyển một đối tượng cho tới khi màu của nó chạm vào một màu xác định khác.



- Một đối tượng sẽ thực hiện một công việc nào đó khi màu sắc của nó chạm vào một màu xác định khác. Ví dụ: nếu màu trắng trên miệng con ngựa chạm vào màu xanh (nước) thì thực hiện việc uống nước...



- Dừng đạn khi nó chạm vào tường màu hồng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Màu () chạm vào màu ()?**.

c. Chuột được nhấn? /Mouse down?

Một vài ví dụ phổ biến thường dùng lệnh Chuột được nhấn?

- Khi nhấn chuột, đối tượng sẽ bay lên một khoảng cách nhất định, và rơi xuống khi không nhấn chuột (ví dụ chú chim trong Flappy bird).



- Khi nhấn chuột, đạn sẽ được bắn ra từ một khẩu súng.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Chuột được nhấn?**.

d. Lệnh Phím () được nhấn?/ Key () Pressed?

Một vài ví dụ phổ biến sử dụng Phím () được nhấn?

- Điều khiển di chuyển của một đối tượng.



- Thay đổi những điều đối tượng nói.



Từ quan sát những ví dụ trên, hãy nêu chức năng của lệnh **Phím () được nhấn?**

e. Lệnh Chỉnh Video thành ()/ Turn Video ()

Lệnh **Chỉnh video thành ()** là lệnh cảm biến nó dùng để tắt mở webcam ở trên máy tính hoặc lật hình ảnh webcam theo chiều ngang, phụ thuộc vào các nhiệm vụ cụ thể.

Các tùy chọn của lệnh:

- **(Bật) On** – cho phép mở webcam máy tính.
- **(Tắt) Off** – tắt webcam máy tính.
- **On-flipped** – cho phép sử dụng webcam nhưng hiển thị video ở chế độ đối xứng theo chiều ngang so với video gốc.

Hãy thực hiện ví dụ và đưa ra nhận định về chức năng của lệnh **Chỉnh video thành ()**.

f. Chỉnh độ mờ video thành () % /Set video transparency to () %

Lệnh này nằm trong nhóm lệnh cảm biến, nó cho phép thiết lập độ trong suốt của video với các mức độ tùy chọn.

Hãy thực hiện ví dụ và đưa ra nhận định về chức năng của lệnh **Chỉnh độ mờ video thành () %**.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Thiết kế trò chơi “Học toán cùng Mèo con” – Tính tổng hai số a và b

Ý tưởng: Trong chương trình sẽ có hai đối tượng trong một cuộc đua “toán học”. Người chơi sẽ đóng vai trò là chú mèo, máy tính sẽ là chú chó (ví dụ như hình).



Chú chó sẽ đưa ra đề bài và nhiệm vụ của chú mèo là phải trả lời đáp án trong thời gian sớm nhất có thể. Nếu chú mèo trả lời đúng, chú mèo sẽ được tiến về phía đích màu

đò 10 bước, nếu sai chú chó sẽ tiến về phía đích 20 bước. Nếu mèo chạm đích trước bạn sẽ là người thắng cuộc, ngược lại bạn thua cuộc.

Gợi ý:

- Việc đầu tiên, bạn hãy khai báo các biến a, b để lưu giá trị các số hạng.
- Chương trình sẽ liên tục gán cho a và b các giá trị ngẫu nhiên để giúp cho mỗi đề bài mà chú chó đưa ra sẽ có các giá trị khác nhau.



- Dùng lệnh **hỏi** `What's your name?` và **chờ** để đưa ra câu hỏi.
- Dùng lệnh **Nếu () thì, còn không thì** để kiểm tra câu trả lời của người chơi (con mèo) xem có đúng hay không.



- Nếu người chơi trả lời đúng sẽ gửi một thông báo **Thông báo** `Mèo đúng` và chú mèo sẽ tiến về phía đích 10 bước. Còn ngược lại chú chó sẽ tiến thêm 20 bước (như hình trên).



- Kiểm tra kết thúc chương trình: Nếu chú mèo chạm đích màu đỏ trước chúng ta sẽ thông báo người chơi thắng cuộc, ngược lại sẽ thông báo thua cuộc và chuyển các hình nền tương ứng.



- Lưu lại chương trình với tên "**Học toán cùng Mèo con**".

C. HOẠT ĐỘNG SÁNG TẠO

1. Chủ đề

Bằng việc sử dụng công cụ lập trình Scratch và các kiến thức đã học (bao gồm cả lịch sử), các bạn hãy lập trình một trò chơi hoặc một đoạn phim ngắn mô tả tóm tắt lại quá trình dẫn đến thắng lợi lịch sử 30/4 giải phóng hoàn toàn miền Nam, thu đất nước thống nhất về một mối.

2. Yêu cầu

➤ Nội dung trò chơi, video khuyến khích sự sáng tạo, vui vẻ và mang tính đậm tính nhân văn và tự hào dân tộc. Không đưa những nội dung trái với thuần phong mỹ tục, không đúng với lịch sử vào chương trình.

➤ Khuyến khích những trò chơi, video có cốt truyện và hướng dẫn cụ thể, rõ ràng, thể hiện nổi bật các mốc lịch sử.

Đối với đoạn video:

- Độ dài tối thiểu 2 phút, tối đa 5 phút.

- Khuyến khích việc sử dụng các hiệu ứng di chuyển đổi tượng, chuyển hình nền, tương tác, giao tiếp giữa các đối tượng trong phim.

Đối với trò chơi:

- Khuyến khích sử dụng tương tác với người chơi theo nhiều cách khác nhau: chuột, bàn phím, âm thanh, camera....

- Khuyến khích sử dụng hiệu ứng di chuyển, tương tác giữa các đối tượng và các hiệu ứng âm thanh.

3. Xây dựng kịch bản



Ý tưởng và kịch bản phải được xây dựng trên cơ sở chủ đề **"Ngày giải phóng miền Nam, thống nhất đất nước 30/4"**.

4. Trình bày dự án

Bạn hãy thực hiện dự án và trình bày dự án trước lớp.

PHẦN 2.

LẬP TRÌNH 3D VỚI KODU GAME LAB

LỜI NÓI ĐẦU

Nội dung “**Lập trình 3D với KODU**” được biên soạn với mục đích giúp học sinh trung học cơ sở sử dụng một cách hiệu quả các lệnh của KODU để tạo ra các hoạt hình hoặc trò chơi với các nhân vật đã được thiết kế sẵn. Các nhân vật được lựa chọn phong phú như KODU, xe tự hành, xe máy, cá, cây cối,... Đồng thời, học sinh có thể lựa chọn những hành vi tương tác giữa các nhân vật.

Tài liệu biên soạn được theo 5 mô đun:

Mô đun 1 - Sử dụng Kodu để tạo cảnh hoạt hình

Mô đun 2 - Thiết kế trò chơi với phần nền phức tạp có 2, 3 nhân vật/ đối tượng

Mô đun 3 - Thiết kế trò chơi có sự điều khiển của người chơi; Lập trình hành vi của nhân vật có điều kiện

Mô đun 4 - Tìm hiểu các lệnh lập trình và giới thiệu sản phẩm với cộng đồng

Mô đun 5 - Luyện tập nâng cao

Mỗi mô đun có cấu trúc như sau:

A. *Hoạt động tìm tòi, khám phá*: Học sinh chủ động khám phá các kiến thức mới trong môn học, thông qua việc thử nghiệm các câu lệnh, chức năng trong Kodu, từ đó đưa ra những nhận xét.

B. *Hoạt động thực hành, ứng dụng*: Học sinh sẽ được luyện tập, củng cố những kiến thức, kỹ năng, phương pháp đã khám phá được ở hoạt động A. Có thêm một số tình huống yêu cầu phải ứng dụng trong điều kiện tương đối mới.

C. *Hoạt động sáng tạo*: Học sinh vận dụng các kiến thức, kỹ năng đã phát hiện được để thực hiện một dự án học tập. Dự án sẽ nêu chủ đề các yêu cầu cần về kịch bản trò chơi, cách thức chơi. Học sinh có thể làm việc theo cặp, nhóm, sau đó trình bày, bảo vệ sản phẩm trước giáo viên và các bạn.

Kodu Game Lab có thể được tải về tại địa chỉ <https://www.kodugamelab.com/> (chọn **Get Kodu**). Tài liệu này hướng dẫn các thao tác với phiên bản 1.4.184.0. Nội dung này đồng thời được số hóa tại <http://digitalskill.vn>. Dưới sự hướng dẫn của các giáo viên, học sinh có thể tự học theo các tài liệu để sáng tạo được nhiều hoạt hình thú vị. Việc đánh giá kết quả học tập được thực hiện qua đánh giá thường xuyên và đánh giá kết quả thực hiện dự án.

Hy vọng rằng “Lập trình 3D với KODU” sẽ giúp các bạn học Tin học một cách chủ động và sáng tạo.

Mô đun 1. SỬ DỤNG KODU ĐỂ TẠO CẢNH HOẠT HÌNH

MỤC TIÊU

- Tìm hiểu được các đặc điểm của trò chơi điện tử.
- Tạo được bối cảnh nền, chỉnh sửa nền, tăng diện tích nền, tạo vùng nền cao như đồi núi, nhà cửa; tạo các vùng thấp như hồ nước.
- Lựa chọn được các đối tượng, nhân vật tham gia hoạt hình (trên cạn, trên không, dưới nước).
- Điều chỉnh được vị trí, hành vi của các đối tượng, nhân vật
- Điều chỉnh được góc quan sát, phóng to, thu nhỏ toàn cảnh.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Tìm hiểu đặc điểm của trò chơi điện tử

Các bạn thử chọn và chơi một trò chơi điện tử. Các bạn hãy thảo luận với nhau, nêu các đặc điểm của trò chơi (đánh dấu * vào ô trống thích hợp và có thể thêm các đặc điểm khác):

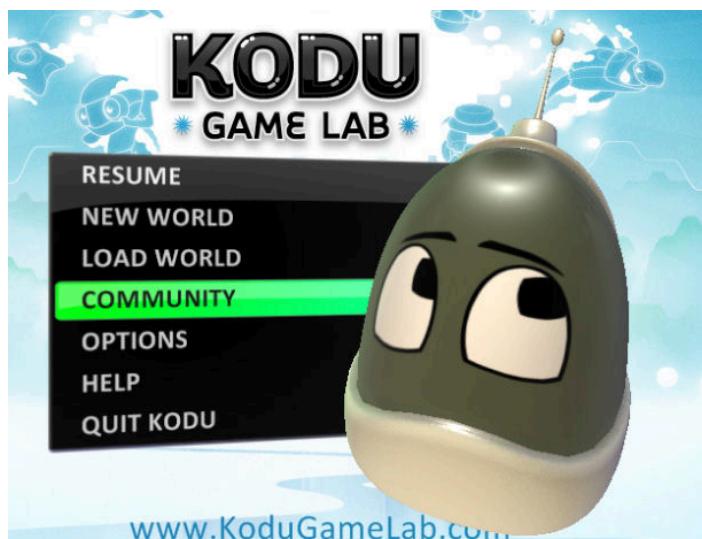
- Trò chơi giúp suy nghĩ, sáng tạo
- Trò chơi tạo hứng thú
- Trò chơi chỉ dành cho trẻ em
- Trò chơi có phần nền (nhà cửa, đồi núi, hồ, sông suối, ...) và có các đối tượng và nhân vật hoạt động và tác động tới nhau
- Trò chơi có thắng, thua, có cho điểm
- Chỉ có các chuyên gia mới tạo được các trò chơi



Bạn có thể sáng tạo ra trò chơi điện tử nhờ phần mềm KODU GAMELAB(gọi tắt là KODU).

2. Chạy và tìm hiểu một số chức năng của chương trình KODU

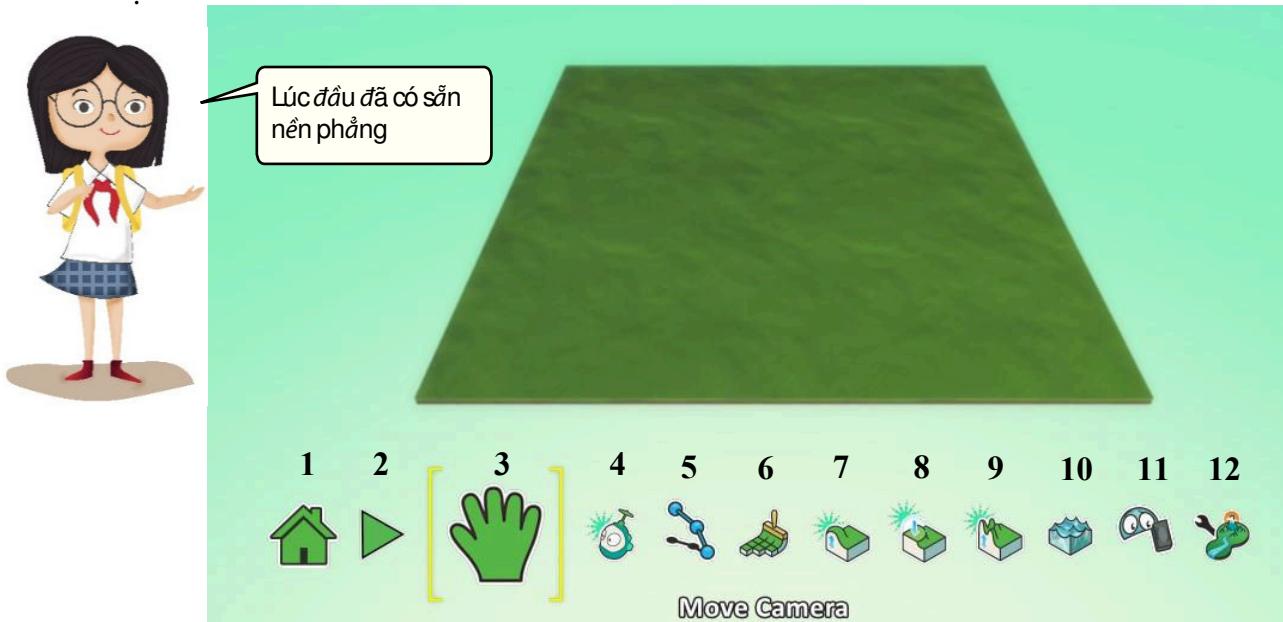
Bạn hãy quan sát màn hình, tìm biểu tượng  của chương trình KODU. Sau đó, bạn hãy chạy chương trình KODU, nhận xét về cửa sổ của KODU và tìm hiểu ý nghĩa các lựa chọn trong KODU.



Resume	Mở chương trình đang được xây	Options	Các tùy chọn
New world	Tạo một chương trình mới	Help	Trợ giúp
Load world	Tải chương trình đã có	Quit Kodu	Thoát khỏi môi trường
Community	Tải chương trình từ Internet		

3. Xây dựng địa hình

a. Chọn menu **NEW WORLD**, quan sát tìm hiểu chức năng các menu ở cửa sổ mới xuất hiện:



Các ký hiệu từ trái sang phải:

- (1) Về menu chính
- (2) Chạy chương trình trò chơi
- (3) Thay đổi góc quan sát của Camera
- (4) Tạo đối tượng và đặc điểm
- (5) Tạo đường dẫn
- (6) Thêm bớt nền
- (7) Tạo núi, thung lũng
- (8) Điều chỉnh độ mấp mô
- (9) Tạo các đỉnh hoặc núi
- (10) Đổ nước
- (11) Xóa đối tượng
- (12) Thay đổi quá trình làm việc

b. Khi bắt đầu tạo trò chơi mới, trên màn hình đã có sẵn bối cảnh nền (hình chữ nhật màu xanh lá cây), hãy sử dụng tính năng  để thử điều chỉnh góc quan sát của camera bằng cách di chuột, kéo thả chuột trái, chuột phải và sử dụng bánh xe chuột. Trao đổi với bạn về các thay đổi đó và ghi lại nhận xét.

c. Thủ sử dụng các tính năng     và thảo luận với bạn về các chức năng được thực hiện.

- Chọn tính năng  (bút vẽ nền), dùng các tính năng phụ kèm theo để chọn kiểu bút; sử dụng chuột trái, chuột phải để thêm, bớt diện tích nền, định màu, tăng giảm kích thước của bút vẽ nền. Trao đổi với bạn về các chức năng được thực hiện.

- Chọn tính năng , sau đó nháy chuột vào địa hình để thay đổi độ cao của nền, tạo đồi núi, nhà cao tầng, thung lũng (có chú ý tới tác dụng chuột trái và chuột phải). Nhận xét những chức năng được thực hiện.

- Chọn tính năng , sau đó nháy chuột để điều chỉnh độ mấp mô của địa hình (thử chuột trái và chuột phải để nâng cao một vùng, hạ thấp độ cao một vùng địa hình). Nhận xét về các chức năng được thực hiện.

- Chọn tính năng , sau đó nháy chuột để tạo các đỉnh cao trên địa hình.

- Sử dụng tổng hợp xen kẽ nhiều tính năng để điều chỉnh tạo ra một khung cảnh nền gồm các đồi núi mấp mô, nhà cao tầng và thung lũng ở giữa.

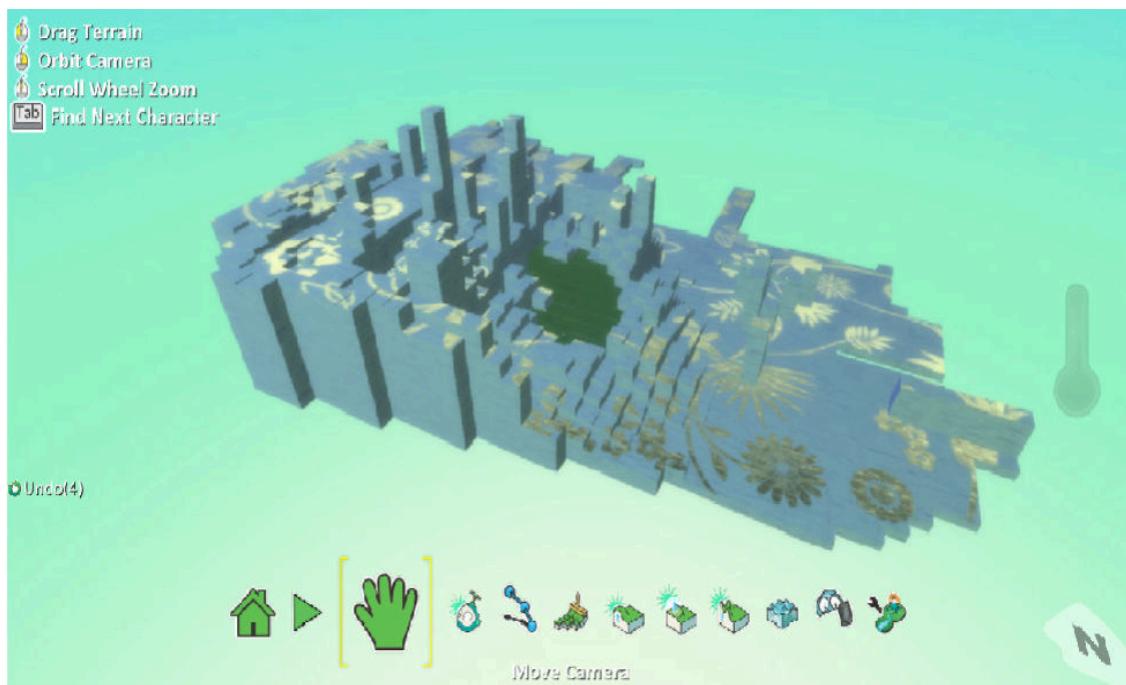
d. Chọn tính năng , sau đó giữ và di chuyển chuột để nhìn toàn cảnh từ các góc quay camera khác nhau. Tự điều chỉnh cảnh nền theo ý thích của mình, nêu nhận xét.

e. Chọn tính năng , sau đó nháy chuột để “đổ nước” vào thung lũng trên địa hình vừa tạo, chú ý các bờ của thung lũng phải khép kín để nước không tràn ra ngoài. Sử dụng chuột trái và chuột phải để điều chỉnh thay đổi mực nước. Ghi lại nhận xét.

4. Lưu chương trình

Hãy lưu lại chương trình của mình theo các bước sau:

- Quay về menu chính bằng cách chọn .
- Chọn **Save my world**.
- Ghi tên chương trình (chẳng hạn bạn Lan Anh ghi là “Hoat canh 1 - Lan Anh”, rồi chọn **Save** (dưới đây là kết quả bối cảnh nền do Lan Anh thiết kế).



- Thoát khỏi chương trình KODU bằng cách chọn “**Exit To Main Menu**”, sau đó chọn “**Quit KODU**”.

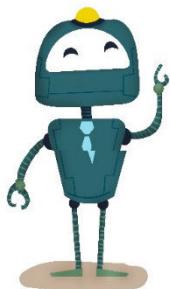
5. Tạo đối tượng, nhân vật cho trò chơi

a. Hãy khởi động lại KODU sau đó mở lại chương trình đã soạn ở tiết trước (chẳng hạn “Hoat canh 1 – Lan Anh”). Chọn chế độ **Edit** để tiếp tục thiết kế chương trình của mình.

b. Chọn đối tượng/ nhân vật cho trò chơi:

- Chọn tính năng , sau đó di chuyển chuột vào một vị trí trên địa hình, nháy chuột phải và chọn **Add object**.

- Lần lượt thêm từng đối tượng theo các loại đã có sẵn dưới đây, tìm hiểu các menu phụ kèm theo rồi xóa đối tượng đó:



Một số đối tượng/nhân vật có thể đưa vào trong hoạt hình.

Kodu	Kodu	Pipe	Ống dẫn
Rock	Tảng đá	Tree	Cây
Underwater	Sinh vật dưới biển	Rover	Xe tự hành
		Apple	Quả táo



Chẳng hạn, chọn  sau đó xuất hiện: 4 kiểu cây khác nhau, có thể chọn 1 trong 4 kiểu đó để đưa vào bối cảnh nền đã thiết kế.

- Tìm hiểu cách xử lý, điều chỉnh đối tượng/ nhân vật theo mẫu xử lý đối tượng



KODU, bạn hãy chọn  và di chuyển chuột tới , nháy chuột phải thì sẽ xuất hiện menu tiếp theo:



Program	Lập trình hành vi	Change size	Thay đổi kích thước
Change Settings	Thay đổi cài đặt	Rotate	Quay đối tượng
Cut	Cắt bỏ	Change height	Điều chỉnh độ cao
Copy	Sao chép		

- Hãy tạo 4 đối tượng: cây, quả táo, KODU, xe tự hành và sắp xếp các đối tượng này lên trên nền địa hình. Điều chỉnh kích thước, góc quay, độ cao của từng đối tượng cho phù hợp.

c. Tìm hiểu cách tạo hành vi cho đối tượng

Đối tượng được gán các tính chất/ hành vi nào đó, bạn có thể chọn hành vi dự kiến cho đối tượng. Chẳng hạn, hãy gán cho KODU hành vi “di chuyển tự do” bằng các bước sau:



+ Chọn , Nhấp chuột phải (Xuất hiện menu mới)

+ Chọn **Program** trong menu tiếp theo, bạn sẽ thấy xuất hiện biểu tượng sau:



+ Chọn sau đó chọn

d. Chạy thử chương trình:

- Bạn hãy chạy thử chương trình bằng cách chọn . Quan sát, trao đổi với các bạn về sự di chuyển của KODU và xe tự hành.

- Lưu lại chương trình trò chơi với tên gọi mới, chẳng hạn, bạn Lan Anh đặt tên là “Hoat canh 2 – Lan Anh”.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Hoạt động 1

Hãy mở lại chương trình vừa soạn và thực hiện các thao tác:

- Mở rộng nền với màu khác.
- Xóa đi một phần nền.
- Thêm một số núi và thung lũng.
- Tăng độ cao của một vài ngọn núi.
- Làm mịn địa hình đồi núi.
- Tạo thêm một thung lũng và đổ nước màu vàng vào thung lũng này.
- Thay đổi góc nhìn của camera, phóng to và thu nhỏ toàn cảnh để quan sát.
- Hãy trao đổi với người khác và ghi lại nhận xét của mình.

2. Hoạt động 2

Hãy xóa đổi tượng “quả táo” và “cây”.

3. Hoạt động 3

Thêm các đối tượng “con sò”, “tảng đá”, hai “cá bay”, “xe máy một bánh”.

4. Hoạt động 4

Tạo hành vi di chuyển (**Move**) cho các đối tượng: “xe tự hành”, “xe máy một bánh”.

5. Hoạt động 5

Tạo hành vi xoay (**Turn**) cho “cá bay” thứ nhất, di chuyển (**Move**) cho “cá bay” thứ hai.

6. Hoạt động 6

Chạy thử chương trình, quan sát và nêu hành vi của các đối tượng vừa được tạo ra, theo bảng sau:

Tên đối tượng	Mô tả hành vi
Cá bay 1	
Cá bay 2	
Xe tự hành	
Con sò	
Tảng đá	
Xe máy một bánh	
KODU	

7. Hoạt động 7

Hãy tự thay đổi cảnh nền và thêm bớt đối tượng, tạo hành vi cho đối tượng, sử dụng chức năng “say” để tạo lời thoại cho đối tượng.

8. Hoạt động 8

Chạy thử và chỉnh sửa, lưu lại chương trình với tên mới “Hoat canh 3 -”, chẳng hạn bạn Lan Anh đặt tên mới là “Hoat canh 3 – Lan Anh”.

C. HOẠT ĐỘNG SÁNG TẠO

1. Chọn hành vi cho đối tượng

Phân công các bạn trong nhóm tìm hiểu về những hành vi có thể lựa chọn cho một đối tượng trong KODU (quả táo, cây, con sò, cá bay...). Các bạn trình bày cho nhau cách tạo hành vi cho đối tượng mà mình đã tìm hiểu, viết các phát hiện của mình vào ô trống theo bảng dưới đây.

Đối tượng chọn	Hành vi dự kiến	Các bước thực hiện
Quả táo		
Cây có 3 cành		
Con sò		
Tảng đá		
Xe tự hành		

2. Thực hiện dự án nhỏ về tạo hoạt cảnh gồm nhiều đối tượng

a. Hãy thảo luận với các bạn, phác thảo một kịch bản về một hoạt cảnh gồm vài đối tượng trên một bối cảnh nền địa hình gồm cả núi, nhà cao tầng, hồ. Trong đó, có một vài đối tượng di chuyển. Mô tả tóm tắt kịch bản theo các nội dung sau:

- Trước hết hãy thảo luận và đưa ra chủ đề của hoạt cảnh; Tên của chương trình là gì?
- Dự kiến về bối cảnh nền gồm những gì, vị trí kích thước ban đầu.
- Dự kiến các đối tượng, đồ vật, cây cối sẽ tham gia vào hoạt cảnh.
- Dự kiến hành vi ban đầu của các đối tượng tham gia.
- b. Tiến hành tạo một chương trình mới
- c. Tạo nền
- d. Tạo các đối tượng và lập trình các hành vi cho các đối tượng theo như đã dự kiến
- e. Chạy thử
- f. Cùng các bạn thảo luận điều chỉnh nền, điều chỉnh về vị trí, kích thước, độ cao và hành vi của đối tượng.
- g. Lưu chương trình theo tên đã dự kiến.
- h. Báo cáo cho thầy/cô giáo về kết quả.

3. Hoàn thiện

Hãy hoàn thiện theo nhận xét của thầy/cô giáo, và lưu lại với tên mới là "Hoat canh 4 -..."

Mô đun 2. THIẾT KẾ TRÒ CHƠI VỚI PHẦN NỀN PHỨC TẠP CÓ 2, 3 ĐỐI TƯỢNG/ NHÂN VẬT

MỤC TIÊU

- Tạo được những phần nền khác nhau về độ cao, lồi lõm, mức độ xù xì.
- Tạo được vài đối tượng khác nhau.
- Tạo các hành vi lựa chọn khác nhau trên bối cảnh nền phức tạp.
- Lập trình tương tác giữa các đối tượng/ nhân vật. Lập trình theo cấu trúc When ... Do

- Thực hiện được dự án nhỏ về trò chơi gồm vài đối tượng với các tương tác phản hồi khác nhau.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Thảo luận

Bạn hãy thảo luận với bạn bè về các câu hỏi:

- Có thể tạo các bối cảnh nền phức tạp như núi, đồi, nhà cao tầng, ao, hồ...không? Thực hiện như thế nào?
- Có thể tạo ra nhiều đối tượng/ nhân vật hơn và tạo kiểu hành vi của từng đối tượng/ nhân vật không?



Một số đối tượng có thể chọn trong KODU.



2. Tiếp tục điều chỉnh hoạt hình

Các bạn cùng mở lại trò chơi đã lưu trong giờ học trước, chạy thử và thảo luận để điều chỉnh và thay đổi, làm cho bối cảnh địa hình phức tạp hơn. Sau đó, sử dụng các menu về địa hình để sửa địa hình sao cho có nhiều đồi, núi, nhà cửa khác nhau và có ít nhất 3 hồ nước. Mở rộng phạm vi diện tích của địa hình. Ghi lại những thay đổi, điều chỉnh của mình.

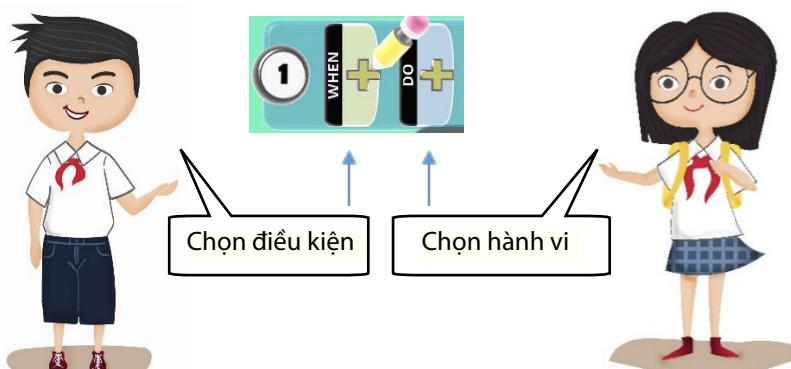
3. Xác định hành vi cho KODU

Bạn tìm hiểu nhân vật KODU, chọn KODU, đưa nhân vật này vào một vị trí trong địa hình. Sau đó, bạn hãy tìm hiểu các khả năng lập trình hành vi của KODU.

a. Trước tiên chọn

b. Chọn nhân vật KODU , đưa KODU tới địa điểm xác định trên nền địa hình, điều chỉnh kích thước, độ cao về vị trí của KODU.

c. Nhấp phím phải chuột, chọn chương trình (**Program**), khi đó xuất hiện **menu** cho phép xác định hành vi của KODU theo điều kiện nào đó:



d. Hãy nhấp đúp vào khi đó xuất hiện bảng **menu** các *hành vi* sau:



Nếu không có điều kiện gì,
Kodu di chuyển tự do.



View	Nhìn	Action	Hành động	Shoot	Bắn
Move	Di chuyển	Settings	Cài đặt	Combat	Chiến đấu
Turn	Xoay	Switch	Chuyển trang	Holding	Cầm giữ
Eat	Ăn	Inline	Chuyển trang	Game	Trò chơi

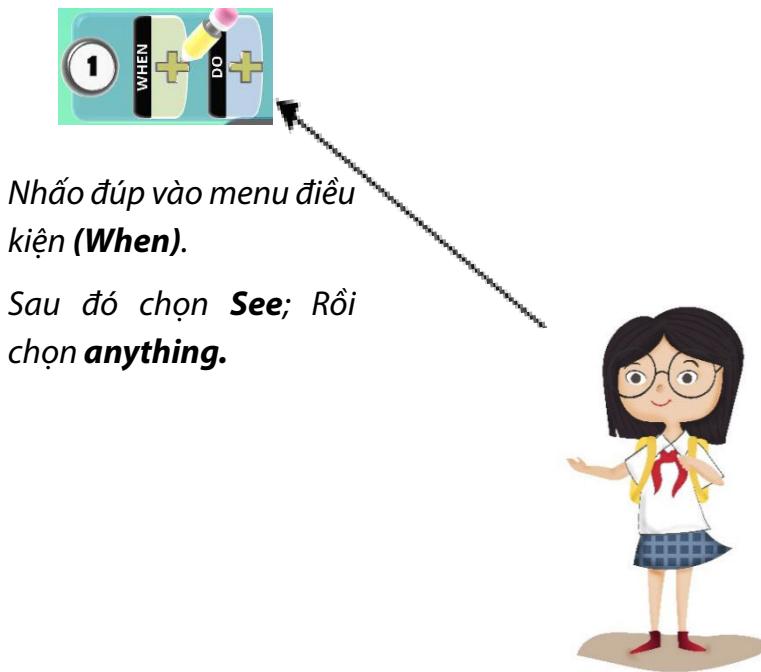
Lưu ý: **Switch** và **Inline** cùng là lệnh dùng để chuyển trang. Khi sử dụng các lệnh này, nhân vật sẽ chuyển sang thực hiện các hoạt động ở một trang khác. Điểm khác biệt ở đây là, khi câu lệnh **Switch** có hiệu lực, nhân vật sẽ ngừng mọi hoạt động còn lại ở trang hiện tại để chuyển sang trang được chọn chuyển tiếp. Ngược lại, khi lệnh **Inline** có hiệu lực, nhân vật sẽ thực hiện các lệnh ở trang chuyển tiếp nhưng đồng thời cũng không bỏ qua các câu lệnh còn lại trên trang hiện tại.

e. Hãy chọn hành vi di chuyển (**Move**) (chưa có điều kiện gì). Chạy thử và trao đổi với bạn bè về hành vi của KODU. Ghi lại nhận xét của mình khi chưa chọn điều kiện mà vẫn chọn hành vi **Move** cho KODU.

4. Thêm đối tượng và chọn điều kiện cho hành vi của đối tượng

Mục trước ta đã lập trình để KODU di chuyển tự do, không theo điều kiện nào. Nay giờ hãy thêm một KODU nữa (gọi là KODU2), lập trình để nhân vật này chuyển động và xác định điều kiện và hành vi: với điều kiện **nhìn thấy (See)** bất kì đối tượng nào (**Anything**) thì ăn luôn! Hãy chạy thử, kiểm tra hành vi của KODU2 này và ghi lại nhận xét.

a. Chọn điều kiện



Các điều kiện có thể xảy ra:



Gamepad	Dùng Gamepad	See	Nhìn thấy	Timer	Sau khoảng thời gian
Keyboard	Dùng bàn phím	Hear	Nghe thấy	Got	Có
Mouse	Dùng chuột	Bumpod	Chạm trán, gấp	Scored	Ghi điểm
Touch	Dùng cảm ứng	End of path	Đi tới cuối đường dẫn	More	Khác

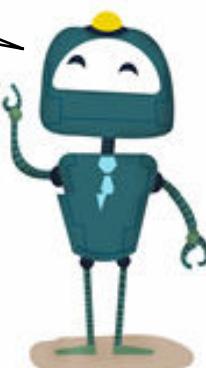
b. Chọn hành vi



- Nháy đúp vào khi đó xuất hiện bảng menu các hành vi sau:



Chọn hành vi
“ăn” (eat)



- Chọn hành vi thích hợp.

5. Chạy thử chương trình

Hãy chạy thử chương trình. Điều chỉnh và lưu chương trình vào máy, lấy tên là “Hoạt canh 5 - ...”.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Thay đổi lệnh

Hãy thử thay đổi hành vi của KODU với một vài điều kiện khác. Chạy chương trình để kiểm tra. Lập bảng theo mẫu dưới đây và điền kết quả:

Điều kiện	Hành vi của KODU	Mô tả kết quả
Thấy bất cứ cái gì (See – anything)	Ăn (thứ đã nhìn thấy)	KODU ăn tất cả những gì mà KODU nhìn thấy

2. Cho KODU chạy theo đường dẫn

Hãy tạo thêm một KODU nữa, cho KODU chạy theo một đường dẫn định trước



(sử dụng chức năng để tạo đường dẫn cho chuyển động của KODU); Ghi lại nhận xét của mình.

3. Thêm đối tượng và hành vi

Lựa chọn một đối tượng khác (chẳng hạn “xe tự hành” hay “cá bay”,..), lập trình để đối tượng này di chuyển, xác định hành vi và chạy thử để kiểm tra. Lập bảng và điền vào bảng về điều kiện, hành vi và mô tả kết quả khi chạy chương trình.

Điều kiện	Đối tượng	Hành vi	Mô tả kết quả

4. Thực hiện theo kịch bản có sẵn

Hãy làm các công việc sau:

- Hãy cùng bạn bè tạo một cảnh hoạt hình theo kịch bản:

TÓM TẮT KỊCH BẢN HOẠT CẢNH

Địa hình: Có một địa hình khá phức tạp, bên phải là núi đồi nhấp nhô, ở giữa là một hồ nước hình bầu dục, bên trái là các tòa nhà cao tầng.

Các đối tượng/nhân vật: Trên địa hình có các đối tượng: KODU đang dạo chơi, có 3 cái cây kiểu dáng khác nhau, có 5 quả táo ở gần các cây; một xe tự hành đang chuyển động tự do trên mặt đất, một vài con sò, sao biển dưới hồ, một “cá bay” đang di chuyển theo một đường dẫn quanh địa hình.

Hành vi quan trọng: Khi di chuyển, KODU thấy táo thì ăn, gặp xe tự hành thì có hành vi **chiến đấu (combat)**, gặp cây thì xoay người và di chuyển sang vị trí khác.

b. Chạy thử chương trình, điều chỉnh nếu cần và so sánh kết quả với nhóm bạn khác.

C. HOẠT ĐỘNG SÁNG TẠO

1. Lập trình hành vi của KODU

Hãy mở tệp “Hoat canh 5”, thêm một số đối tượng trong đó có khoảng trên 15 quả táo, đổi hành vi của KODU2: nếu gặp quả táo thì phải nhảy lên để tránh và đi tiếp.



Thảo luận với các bạn trong nhóm, trong cặp để:

- Xây dựng kịch bản hoạt hình nhỏ với chủ đề tùy ý, trong đó có các đối tượng trên cạn, dưới nước, trên không khác nhau.
- Với từng đối tượng, tạo hành vi theo điều kiện nào đó.
- Chạy thử chương trình, điều chỉnh kịch bản và chương trình.
- Lưu lại với tên là “Hoat canh 6 -....”

2. Tạo kịch bản đơn giản

a. Thảo luận và nghĩ ra một kịch bản về hoạt hình, mô tả địa hình, xác định đối tượng, cần có cả KODU, lập trình hành vi, trong đó cần có sự di chuyển của KODU và của các đối tượng khác, hành vi của đối tượng có thể không cần điều kiện gì và hành vi theo một điều kiện nào đó.

b. Viết tóm tắt kịch bản vào vở.

3. Viết chương trình

- a. Lập chương trình theo kịch bản.
- b. Trình diễn sản phẩm trước các bạn thuộc nhóm khác. Trao đổi kinh nghiệm về cách thiết kế. Điều chỉnh hoàn thiện sản phẩm của nhóm mình.
- c. Lưu chương trình (nên đặt tên chương trình có ý nghĩa theo chủ đề đã chọn) .

Mô đun 3. THIẾT KẾ TRÒ CHƠI CÓ ĐIỀU KHIỂN, LẬP TRÌNH HÀNH VI CỦA ĐỐI TƯỢNG CÓ TƯƠNG TÁC

MỤC TIÊU

- Tiếp tục khám phá thêm các khả năng của các đối tượng trong thế giới của KODU, thiết kế địa hình tinh tế và phức tạp.
- Tiếp tục thử nghiệm, phát hiện thêm các hành vi của một số đối tượng trong thế giới KODU.
- Lập trình cho phép điều khiển hành vi của đối tượng/nhân vật bằng bàn phím.
- Thực hiện dự án thiết kế trò chơi có điều khiển hành vi của đối tượng thông qua sử dụng bàn phím của người chơi.
- Trao đổi với bạn bè qua cộng đồng KODU, đăng tải sản phẩm lên diễn đàn của cộng đồng.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Tìm hiểu, phân tích trò chơi có sẵn

Bạn hãy cùng với các bạn làm các công việc sau:

a. Tìm hiểu một trò chơi được tạo bởi KODU lấy từ nguồn cộng đồng trên Internet. Khởi động trò chơi, nhận xét về:

- Thiết kế địa hình thế nào?
- Các đối tượng và nhân vật gồm những gì?
- Hành vi ban đầu của các đối tượng, nhân vật?
- Tương tác giữa các đối tượng?
- Việc điều khiển đối tượng bằng bàn phím, bằng chuột như thế nào?
- Đánh giá việc chơi như thế nào?
- Khi nào kết thúc trò chơi?

Để tham khảo một trò chơi được tạo bởi KODU, các bạn hãy truy cập vào địa chỉ: <http://kodugamelab.com> và vào phần Worlds.



- b. Bạn có thể tạo các bối cảnh nền phức tạp giống như trò chơi này không, sau đó làm mịn và hoàn thiện lại cảnh địa hình như núi, đồi, nhà cao tầng, ao, hồ... không? Thực hiện như thế nào?
- c. Có thể tạo ra nhiều đối tượng/ nhân vật hơn và tạo kiểu hành vi của từng đối tượng theo một điều kiện nào đó không?

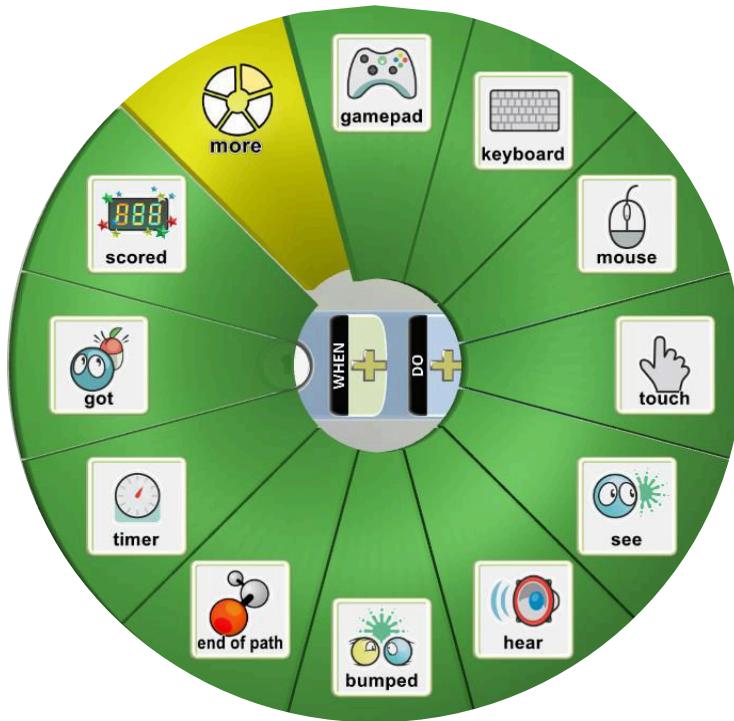
2. Tìm hiểu lệnh chọn hành vi cho KODU

- a. Bạn và các bạn tìm hiểu KODU và một vài đối tượng khác
- b. Bạn lập trình hành vi ban đầu cho KODU (chưa theo điều kiện nào cả). Thủ chạy chương trình và quan sát hành vi của KODU.
- c. Lập trình tương tự các đối tượng khác. Chạy thử và kiểm tra hành vi ban đầu của các đối tượng này có đúng theo ý lập trình của mình không.



3. Tìm hiểu về các điều kiện có thể xảy ra cho KODU

Bạn hãy tìm hiểu về các điều kiện có thể xảy ra cho KODU và giải thích ý nghĩa của các điều kiện đó.



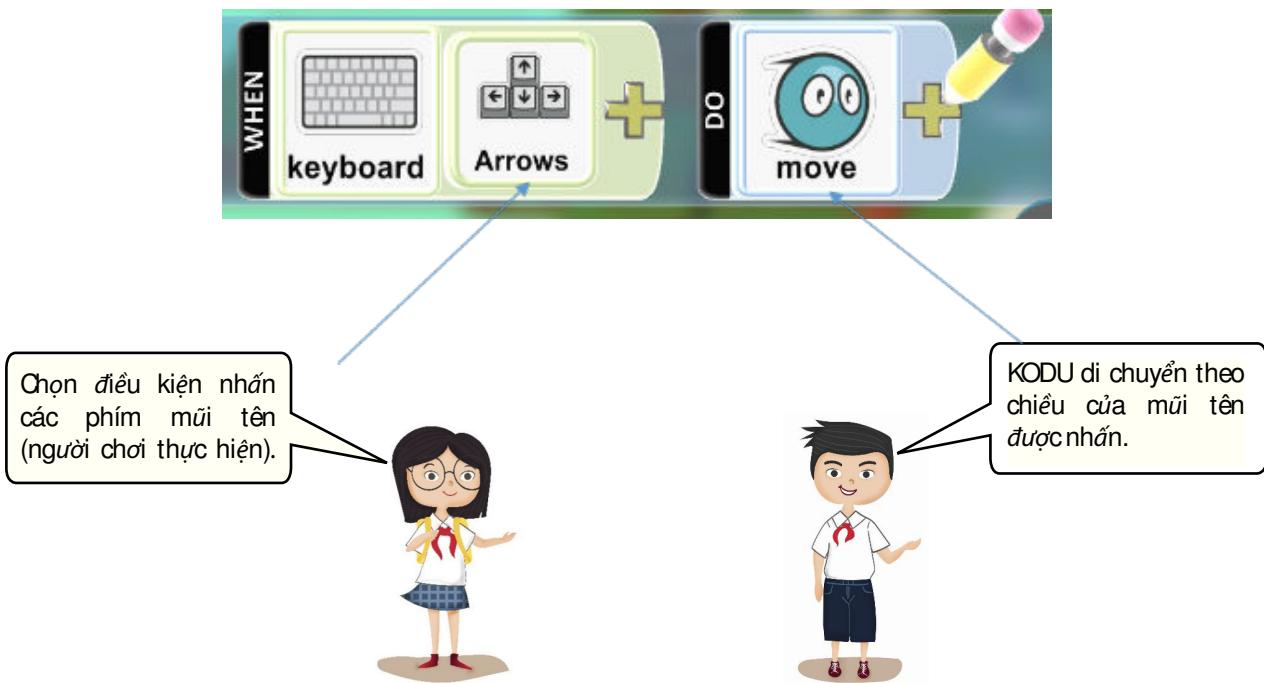
4. Điều khiển KODU bằng bàn phím

Hãy tìm hiểu cách lập trình cho KODU để có thể điều khiển KODU bằng bàn phím:

a. Chọn điều kiện tác động KODU từ bàn phím (các hành vi của KODU sẽ tùy theo các thao tác sử dụng bàn phím của người chơi), theo dãy thao tác sau:

- Chọn KODU
- Chọn Program
- Chọn điều kiện là sau đó chọn

b. Thử thêm lệnh điều chỉnh hành vi của KODU, hãy cho KODU di chuyển tự do, nhưng gặp một đối tượng bất kì trên đường thì xoay người và chuyển hướng đi. Thảo luận với bạn bè: cần chọn lệnh gì cho điều kiện? Chọn lệnh gì cho hành vi?



Hãy chạy thử chương trình, đóng vai người sử dụng, dùng các phím mũi tên để điều khiển KODU di chuyển.

Thảo luận và ghi lại nhận xét của mình vào vở.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Thay đổi hành vi của KODU với một vài điều kiện khác

Bạn hãy lập trình hoạt động cho KODU như sau:

- Nếu người chơi bấm **phím A** thì KODU di chuyển.
- Nếu người chơi bấm **phím B** thì KODU quay sang trái.

Tham khảo các hình vẽ sau về các lệnh trên hình ảnh trực quan:



2. Thủ chạy kiểm tra

Hãy thử chạy và nhấn giữ **phím A** rồi sau đó nhấn **phím B** để kiểm tra khả năng điều khiển KODU từ bàn phím.

3. Lập chương trình điều khiển KODU

Hãy tìm hiểu và lập chương trình để KODU thực hiện các thao tác theo điều khiển của người chơi và theo đoạn kịch bản sau:

- a. Tạo thế giới mới, tạo địa hình mới và đưa vào đó các đối tượng xe tự hành, thêm 10 quả táo, một KODU.
- b. Lập chương trình để KODU di chuyển theo chiều điều khiển mũi tên.
- c. Người chơi di chuyển KODU bằng các phím mũi tên để ăn các quả táo. Mỗi lần KODU gặp quả táo (điều kiện) thì KODU sẽ ăn (hành vi) quả táo đó.
- d. Người chơi sẽ cho KODU di chuyển và ăn hết cả số táo đã có trên nền địa hình.
- e. Lưu chương trình và lấy tên là “Tro Choi 1”.

Ghi lại nhận xét của mình về kết quả lập trình trò chơi.

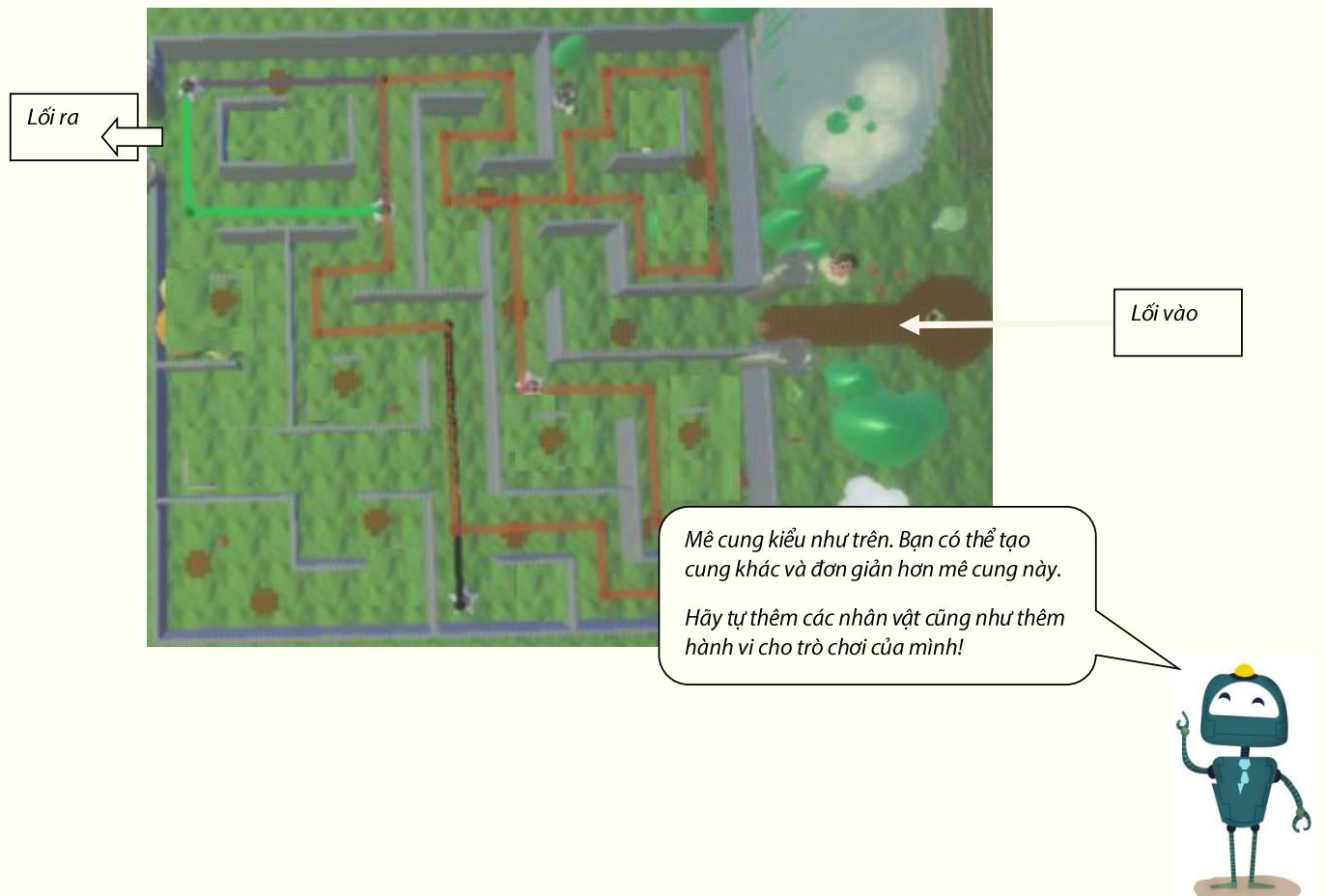
4. Lựa chọn một đối tượng khác và lập trình để đối tượng di chuyển theo đường dẫn cho trước

Hãy lựa chọn một đối tượng khác (chẳng hạn “xe tự hành”), lập trình để đối tượng này di chuyển theo đường dẫn cho trước.

- a. Lập trình cho xe tự hành: nếu gặp KODU thì bắt giữ KODU. Sau đó, chạy thử để quan sát kết quả.
- b. Hãy lập trình để người chơi điều khiển KODU, khi thấy xe tự hành thì tránh để không có va chạm với xe tự hành.
- c. Chạy thử chương trình và điều khiển KODU chạy tránh xe tự hành. Điều chỉnh chương trình và so sánh kết quả với nhóm bạn khác. Ghi nhận xét vào vở.

5. Xây dựng trò chơi mới theo kịch bản tóm tắt

- a. Địa hình bao gồm:
 - Một “mê cung” gồm tường thành có nhiều lớp, chỉ có một cổng vào và một lối ra.
 - Ở trung tâm mê cung có một đối tượng bị giữ (chẳng hạn một con sò mang theo một viên ngọc).
 - Bên ngoài cổng vào là KODU.
 - Dọc đường vào mê cung có một vài quả táo.
- b. KODU tìm cách vào cổng và đi dọc theo tường thành của mê cung, ăn hết táo và đồng thời tìm cách vào trung tâm để lấy con sò và mang ra ngoài.
- c. Khi đưa con sò ra được bên ngoài là thành công.



C. HOẠT ĐỘNG SÁNG TẠO

1. Sáng tạo trò chơi mới

Thảo luận với các bạn theo nhóm hoặc theo cặp để:

a. Xây dựng kịch bản một trò chơi, trong đó có địa hình khá phức tạp, có hồ nước, đồi núi, có các đối tượng trên cạn, dưới nước, trên không.

- Tự nghĩ ra chủ đề.
- Dự kiến địa hình ban đầu.
- Dự kiến các đối tượng sẽ tham gia trong trò chơi.

Với từng đối tượng/ nhân vật, tạo hành vi theo điều kiện xác định nào đó trình bày theo mẫu ở bảng sau:

STT	Tên đối tượng/ nhân vật	Mô tả điều kiện	Mô tả hành vi
1			
2			
..

b. Lập trình để điều khiển hành vi của đối tượng từ bàn phím (di chuyển, quay, nhảy, tránh...).

STT	Đối tượng	Lệnh điều khiển	Hành vi
1			
2			
..

c. Lập trình cho hành vi của đối tượng nhờ điều khiển chuột của người chơi.

d. Chạy thử và điều chỉnh kịch bản và điều chỉnh chương trình.

e. Lưu lại thành file với tên thích hợp.

2. Trình diễn sản phẩm

Trình diễn sản phẩm trước các bạn thuộc nhóm khác. Trao đổi kinh nghiệm về cách thiết kế. Điều chỉnh hoàn thiện sản phẩm của nhóm mình.

Mô đun 4. TÌM HIỂU CÁC LỆNH LẬP TRÌNH VÀ GIỚI THIỆU SẢN PHẨM VỚI CỘNG ĐỒNG

MỤC TIÊU

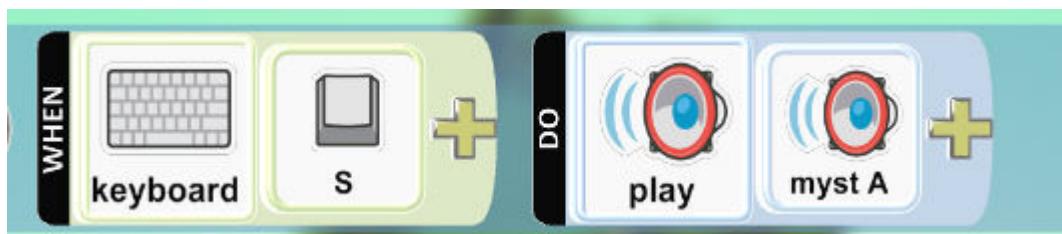
- Tiếp tục khám phá thêm các khả năng của các đối tượng trong thế giới của KODU, thiết kế địa hình tinh tế và phức tạp. Tiếp tục thử nghiệm, phát hiện thêm các hành vi của một số đối tượng trong thế giới KODU.
- Tiếp tục lập trình cho phép điều khiển hành vi của đối tượng/ nhân vật bằng bàn phím.
- Xây dựng và thực hiện dự án thiết kế trò chơi có cho điểm.
- Có thể xử lý âm thanh nền trong khi tạo trò chơi.
- Trao đổi với bạn bè qua cộng đồng KODU, đăng tải sản phẩm lên diễn đàn của cộng đồng.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Chèn âm thanh, nhạc vào khung cảnh/ trò chơi

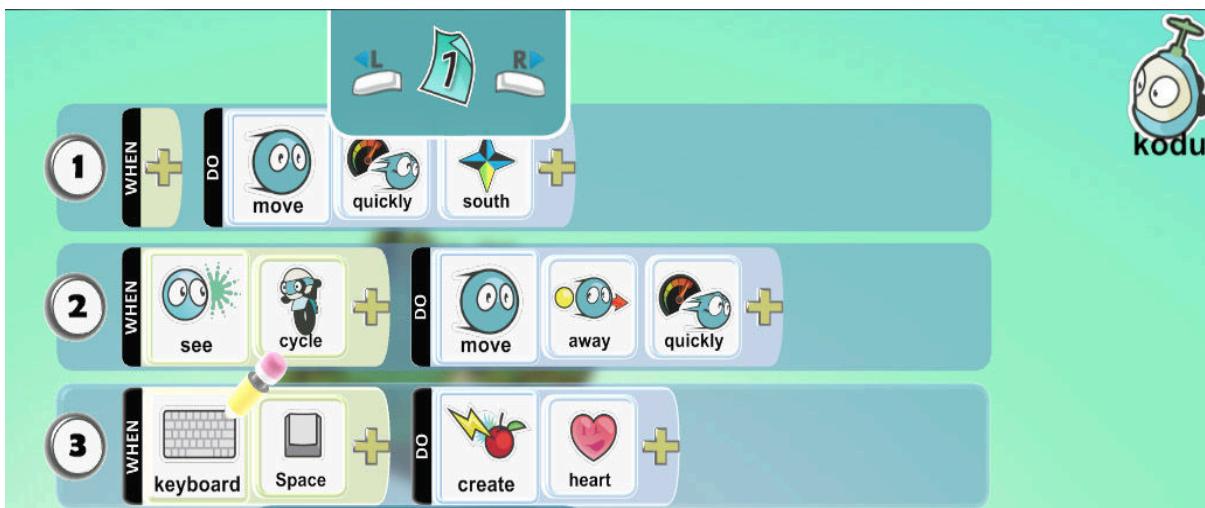
Thông thường, có thể đưa vào khung cảnh của hoạt hình hoặc trò chơi nhạc nền phù hợp.

Bạn hãy thử tìm hiểu cách đưa một đoạn nhạc nền khi người sử dụng nhấn vào phím kí tự  theo các bước thể hiện như sau:



2. Tìm hiểu các trang

Với một trò chơi, có thể soạn nhiều trang lệnh đối với một đối tượng/ nhân vật. Với mỗi trang có nhiều dòng lệnh, chẳng hạn với KODU, có trang 1 với một vài dòng lệnh đầu như sau:



a. Hãy quan sát các dòng lệnh, giải thích cho bạn bè ý nghĩa các dòng lệnh.

b. Thủ tạo trò chơi mới có chứa các đối tượng trong các dòng lệnh và lập chương trình cho KODU theo các dòng lệnh trên và chạy thử để kiểm tra các dòng lệnh của bạn có được thể hiện đúng như những hành vi bạn mong muốn không.

3. Tạo trang mới

a. Tạo trang 2 bằng cách nhấn phím **Tab** trên bàn phím. Bạn thử soạn dòng lệnh như sau:

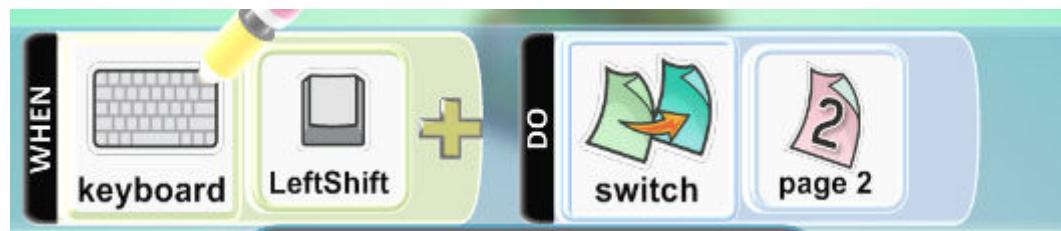


b. Phân tích ý nghĩa dòng lệnh trên.

Dòng lệnh cho phép kết thúc trò chơi sau 4 giây!



c. Như vậy, khi đang thực hiện các dòng lệnh ở trang 1, để thực hiện chương trình ở trang 2, cần có lệnh chuyển trang. Để chuyển trang, hãy soạn dòng lệnh ở trang 1: nếu nhấn phím kí tự **Shift** thì sẽ chuyển sang trang 2, theo mẫu sau:



Chạy thử và nhấn phím **Left Shift** (phím Shift bên trái bàn phím) và quan sát kết quả. Ghi lại nhận xét của mình vào vở.

B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Luyện tập các lệnh

Hãy tạo tập tin mới, tạo địa hình, các đối tượng khác nhau và luyện tập các lệnh sau:

- Điều khiển cho KODU di chuyển bằng các phím mũi tên
- Cho KODU nhảy lên khi bấm phím cách
- Cho KODU tránh sang bên khi gặp cây
- Cho xe mô tô di chuyển tự do
- Cho xe mô tô thay đổi màu sắc liên tục (xanh, đỏ, tím, vàng, đen...)
- Nếu bấm phím **S**, thì xe tự hành bắn súng
- Nếu bấm phím **Left Shift** thì kết thúc trò chơi (tức là chuyển sang trang 2)
- Chuyển sang lập trình trang 2 với nội dung: chờ 8 giây, sau đó kết thúc trò chơi

2. Phân tích các yếu tố của trò chơi

Hãy quan sát các mô tả bằng hình ảnh trong trò chơi đơn giản dưới đây.

- Hoạt cảnh đầu tiên của trò chơi:

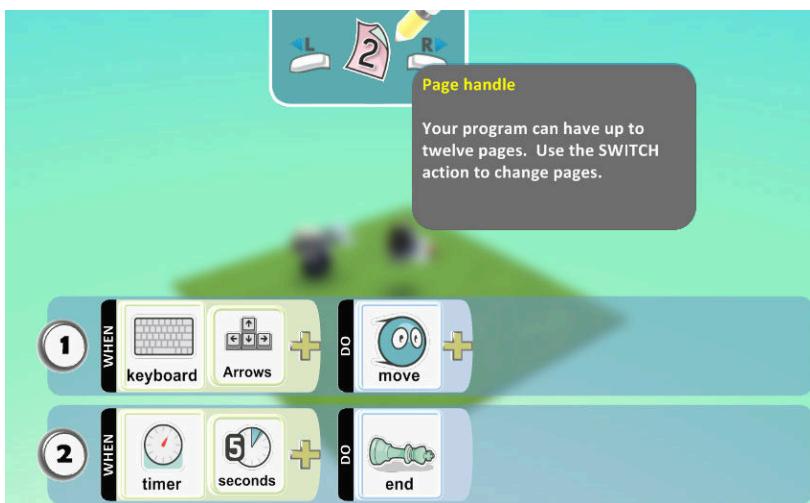


b. Lệnh cho KODU

Trang 1



Trang 2



c. Lệnh cho Mô tô

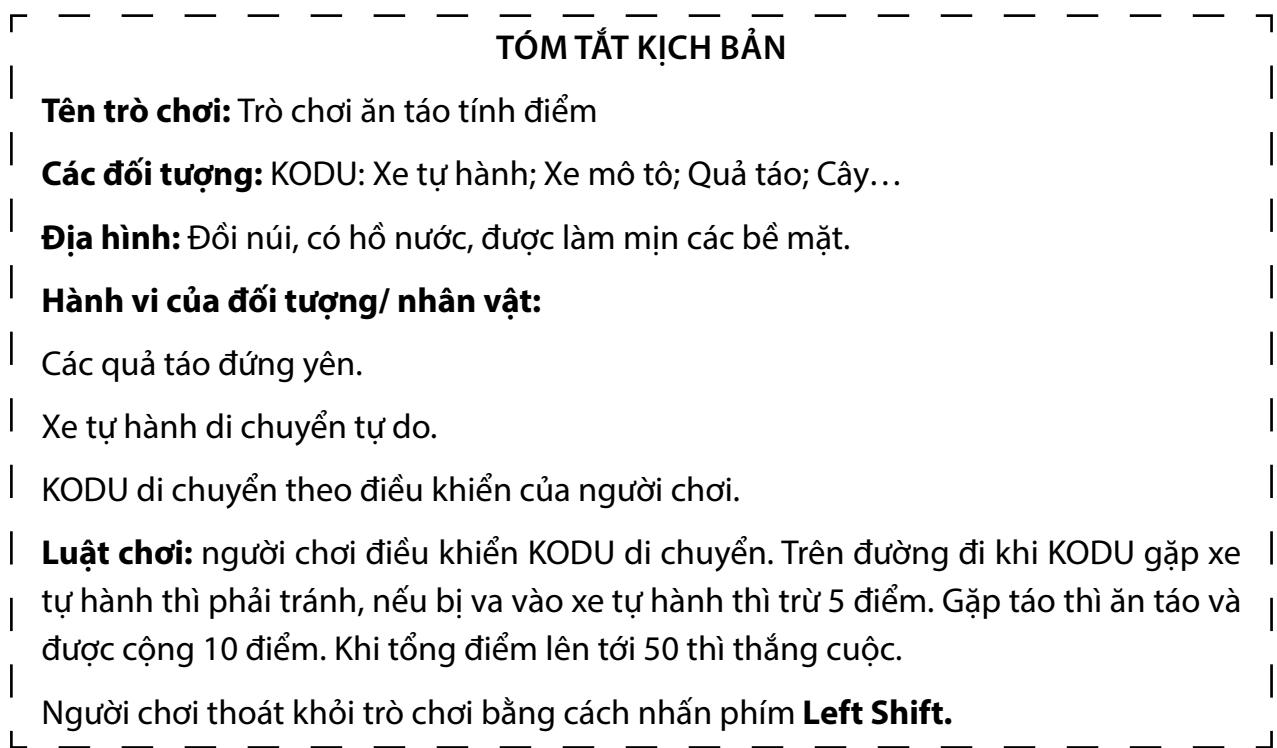


d. Dựa vào kết quả quan sát, trả lời các câu hỏi sau:

- Có các đối tượng nào trong trò chơi?
- Người chơi có thể điều khiển KODU như thế nào?
- Xe mô tô có hành vi thay đổi thế nào?
- Khi muốn kết thúc trò chơi thì người chơi làm thế nào?

3. Xây dựng kịch bản trò chơi

Hãy thảo luận với bạn để tạo trò chơi đơn giản theo kịch bản sau (hoặc tự tạo ra trò chơi khác, có viết kèm theo kịch bản của mình)



C. HOẠT ĐỘNG SÁNG TẠO

1. Hãy về menu chính, sau đó tải xuống (download) một trò chơi trên Internet.
2. Hãy vào chế độ chỉnh sửa (EDIT) để kiểm tra hệ thống lệnh cho các đối tượng/ nhân vật trong trò chơi.

Chú ý tới các yếu tố sau:

- a. Đặc điểm của địa hình
- b. Các đối tượng/ nhân vật được xuất hiện trên nền địa hình
- c. Hành vi ban đầu của các đối tượng/ nhân vật
- d. Quan hệ tương tác giữa các đối tượng/ nhân vật

- e. Tương tác với người chơi
 - f. Luật chơi
3. Thủ chơi và kiểm tra lại các lệnh được thực hiện như thế nào
4. Hãy thay đổi một số lệnh cho các đối tượng để có được trò chơi với các hành vi mới của các đối tượng, nhân vật và các luật chơi mới.

a.

Đối tượng	Hành vi mới của đối tượng	Lập trình

b.

Luật chơi mới	Cách lập trình

5. Sáng tạo trò chơi theo nhóm
- a. Xác định kịch bản ban đầu
 - Nêu rõ ý tưởng trò chơi
 - Xác định địa hình
 - Lựa chọn đối tượng/nhân vật chơi và bố trí vị trí ban đầu cho chúng.
 - Xác định hành vi của các đối tượng/ nhân vật.
 - Mô tả tương tác giữa các đối tượng/ nhân vật.
 - Xây dựng luật chơi.
 - Xác định nhạc nền.
 -
 - b. Phân công trong nhóm từ một đến hai bạn phụ trách các phần việc sau:
 - Xây dựng địa hình theo kịch bản.
 - Lập trình cho hành vi của các đối tượng.
 - Lựa chọn nhạc nền.
 -
 - c. Khớp các sản phẩm để tạo trò chơi hoàn chỉnh.
 - d. Chơi thử và hoàn thiện.
 - e. Trình bày, trao đổi trò chơi với các bạn trong lớp và với các bạn khác trên cộng đồng KODU thông qua mạng Internet.

MÔ ĐUN 5. LUYỆN TẬP NÂNG CAO

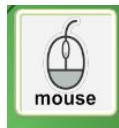
MỤC TIÊU

- Bổ sung thêm cách dùng một số lệnh mới trong KODU.
- Tiếp tục khám phá thêm các khả năng của các đối tượng trong thế giới của KODU; thiết kế địa hình tinh tế và phức tạp. Tiếp tục thử nghiệm, phát hiện thêm các hành vi của một số đối tượng trong thế giới KODU.
- Tiếp tục lập trình cho phép điều khiển hành vi của đối tượng/ nhân vật bằng bàn phím và chuột.
- Xây dựng và thực hiện dự án thiết kế trò chơi có sự thi đua và cạnh tranh giữa hai nhân vật.
- Trao đổi với bạn bè qua cộng đồng KODU, đăng tải sản phẩm lên diễn đàn.

A. HOẠT ĐỘNG TÌM TÒI, KHÁM PHÁ

1. Tìm hiểu việc điều khiển đối tượng bằng chuột thông qua khám phá ý nghĩa các biểu tượng sau:

a.



b.



c.



d.



e.



Có thể dùng chuột điều
kiển hành vi đối tượng!

2. Tạo các đối tượng theo mô tả tranh sau



3. Lập trình cho KODU theo mô tả sau

1. WHEN keyboard DO move, forward

2. WHEN keyboard Space DO shoot

3. WHEN shot hit blimp DO score green points

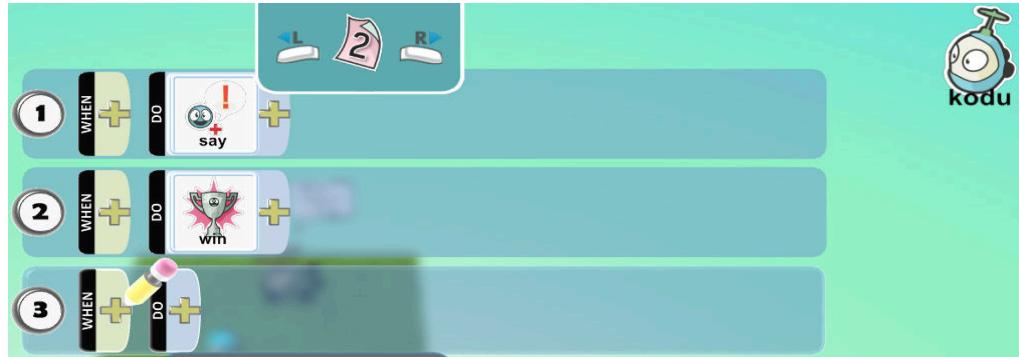
4. WHEN shot hit anybot DO score green points

5. WHEN shot hit anything DO score green points

6. WHEN scored DO red above 100 points win red Player1

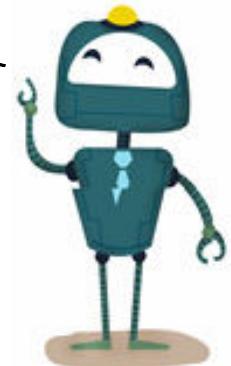
7. WHEN scored DO red above 100 points switch page 2

8. WHEN + DO +



4. Tìm hiểu và thảo luận việc tổ chức chơi hai người, thi đua tính điểm, ai đạt được 100 điểm trước thì thắng cuộc, ghi lại nhận xét.

Có thể tạo trò chơi có hai hoặc nhiều người chơi



5. Tìm hiểu lệnh Inline

a. Lập trình trò chơi, tạo địa hình và đối tượng, trong đó trang 1 có nội dung sau:

Chạy thử và giải thích các lệnh ở trang 1.



b. Lập trình trang 2 theo nội dung sau:



c. Sửa lại các lệnh ở trang 1 như sau:



d. Chạy thử và so sánh với việc thực hiện chương trình ở mục a:

Nêu nhận xét ý nghĩa của các dòng 1, 2, 3, 4, 5, 6,...

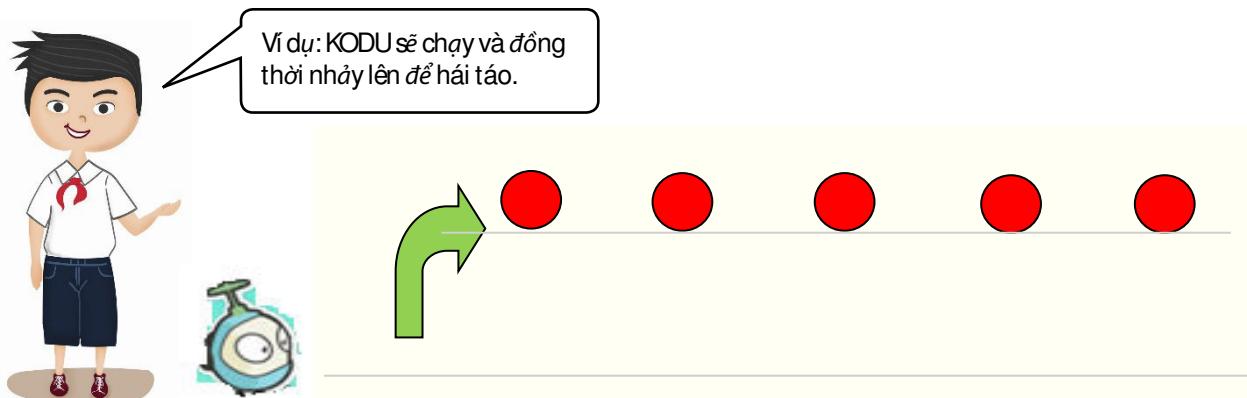
B. HOẠT ĐỘNG THỰC HÀNH, ỨNG DỤNG

1. Hãy nghĩ cách lập trình để thực hiện các việc sau

a. Chèn thêm âm thanh nào đó.

b. Thêm lời thoại cho nhân vật.

- c. Tính điểm cho mỗi người chơi (đại diện cho một đối tượng trong trò chơi).
- d. So sánh điểm của những người chơi.
- e. Xác định công việc cần làm hoặc hành vi của đối tượng nào đó nếu điểm vượt một số cố định (chẳng hạn 100 điểm).
- f. Cho phép người chơi điều khiển đối tượng chạy, nhảy, tránh, ăn, ... khi gặp tình huống thích hợp.



- g. Chuyển trang tiếp theo.
- h. Khi đã đạt được kết quả nào đó thì chuyển trang sau và dừng cuộc chơi.

2. Phân tích các lệnh lập trình cụ thể

- a. Hãy quan sát hình ảnh các lệnh, giải thích ý nghĩa các lệnh.



- b. Hãy quan sát hình ảnh các lệnh, giải thích ý nghĩa các lệnh.



3. Nhớ lại những gì đã làm được

Hãy nêu mức độ thực hiện được các công việc kể ra ở dưới đây (theo 3 mức độ: C. chưa thành thạo, B. thành thạo, A. rất thành thạo):

- Sử dụng menu trong xử lý địa hình.
- Nắm được các đối tượng/ nhân vật được sử dụng trong môi trường KODU, chọn và cài đặt các đối tượng này trên nền địa hình đã xác định (về vị trí, độ lớn,...)
- Hiểu được các hành vi có thể chọn cho KODU, Xe tự hành, Xe mô tô một bánh, Sao biển, Quả táo.
- Nêu các điều kiện có thể lựa chọn khi lập trình hành vi cho đối tượng.
- Tạo được trò chơi trong đó có sự tương tác giữa các đối tượng.
- Tạo được trò chơi có điều khiển của người chơi có ghi điểm, hoặc có tính thời gian.
- Tạo được trò chơi có hai người điều khiển đối tượng và tuân theo luật thắng thua.

C. HOẠT ĐỘNG SÁNG TẠO

Hãy về menu chính, sau đó tải xuống (**Download**) một trò chơi nữa trên Internet

1. Hãy vào chế độ EDIT để kiểm tra hệ thống lệnh cho các đối tượng/ nhân vật trong trò chơi.

Chú ý tới các yếu tố sau:

- Đặc điểm của địa hình
- Các đối tượng/ nhân vật được xuất hiện trên nền địa hình
- Hành vi ban đầu của các đối tượng/ nhân vật
- Quan hệ tương tác giữa các đối tượng/nhân vật
- Tương tác với người chơi

g. Luật chơi

2. Thủ chơi và kiểm tra lại các lệnh được thực hiện như thế nào, nếu kịch bản của trò chơi này.

3. Hãy thay đổi một số lệnh cho các đối tượng để tạo trò chơi với các hành vi và luật chơi mới.

a.

Đối tượng	Hành vi mới của đối tượng	Lập trình

b.

Luật chơi mới	Cách lập trình

4. Hãy xây dựng trò chơi theo chủ đề

Hai người chơi thi làm Toán, mỗi bạn phụ trách điều khiển một đối tượng, máy sẽ ra các phép tính ở dạng các phép toán (tổng số 20 phép toán), mỗi đối tượng mang một biển hiệu phép toán máy ra cho mình, người chơi phụ trách đối tượng nào sẽ điều khiển đối tượng đó chạy tới quả táo có ghi kết quả tương ứng phép toán đó.

Người chơi nào tính đúng nhiều phép toán nhất sẽ được thưởng.

5. Hãy cùng bạn sáng tạo trò chơi mới

a. Xác định kịch bản ban đầu (chủ đề tự do).

- Nêu rõ ý tưởng trò chơi.
- Xác định địa hình.
- Lựa chọn từ 5 đến 6 đối tượng/ nhân vật chơi và bố trí vị trí ban đầu cho chúng.
- Xác định hành vi ban đầu của các đối tượng/ nhân vật.
- Mô tả các hành vi tương tác giữa các đối tượng/ nhân vật.
- Xác định từ 3 đến 4 người chơi.
- Xây dựng luật chơi.
- Xác định nhạc nền.
- Nội dung khác nếu cần.

b. Phân công trong nhóm mỗi bạn hoặc hai bạn phụ trách các phần việc:

- Xây dựng địa hình theo kịch bản.

- Lập trình cho hành vi của các đối tượng.

- Lập trình việc thực hiện lệnh điều khiển đối tượng cho từng người chơi (chẳng hạn: người chơi ra lệnh cho đối tượng chạy, nhảy hay bắn, bắt giữ, chạy theo đối tượng khác...).

- Lựa chọn nhạc nền.

- Nhiệm vụ khác nếu cần.

c. Cả nhóm khorp các sản phẩm để bước đầu tạo trò chơi hoàn chỉnh.

d. Chơi thử và hoàn thiện trò chơi.

e. Trình bày, trao đổi trò chơi của mình với các bạn trong lớp và với các bạn khác trong cộng đồng KODU qua Internet.

Phụ lục 1: MẪU MÔ TẢ HOẠT HÌNH THIẾT KẾ BẰNG SCRATCH VÀ KODU

Tên trường:	Tên học sinh:
Lớp:	
Tên hoạt hình:	Em dọn nhà giúp mẹ
Mục tiêu của hoạt hình:	<p>Mô tả tổng quát ý tưởng của trò chơi, các yêu cầu người chơi phải thực hiện</p> <p>Trò chơi nhằm nâng cao ý thức vệ sinh chung và bảo vệ môi trường Nhằm rèn luyện khả năng quan sát và phản xạ cho người chơi Người chơi cần điều khiển đối tượng thực hiện hành động nhặt rác và phân loại rác, bỏ rác đúng nơi quy định</p>
Bối cảnh của hoạt hình:	<p>Mô tả các thành phần của bối cảnh game</p> <p>Ví dụ: Khi đi học về em thấy trong nhà mình có rất nhiều rác, đồ chơi do các em nhỏ chơi và vứt bừa bãi ra nhà. Hãy giúp mẹ dọn dẹp và phân loại rác nhé</p>
Các nhân vật, các đối tượng trong hoạt hình	<p>Liệt kê các đối tượng, các nhân vật trong game và mục đích của các đối tượng, nhân vật đó</p> <p>(Ví dụ)</p> <ul style="list-style-type: none"> - Rác các loại (vỏ chai, giấy vụn, đồ ăn, ...) - Mẹ - Các em nhỏ. - Đối tượng chính (người chơi) ...
Mô tả quy tắc của hoạt hình	<p>Làm thế nào để chiến thắng hoặc kết thúc trò chơi? Làm thế nào để cộng điểm/bị trừ điểm? Thời gian tối đa của hoạt hình là bao lâu?...</p> <p>Dùng chuột để chọn và phân loại rác Cộng, trừ điểm: Mỗi lần phân loại đúng người chơi sẽ được cộng thêm 1 điểm, phân loại sai sẽ bị trừ 1 điểm Hoạt hình không giới hạn thời gian Hoạt hình sẽ lưu lại kỷ lục của người chơi.</p>

Phụ lục 2: BẢNG TIÊU CHÍ ĐÁNH GIÁ HOẠT HÌNH THIẾT KẾ

BẢNG SCRATCH VÀ KODU

TIÊU CHÍ		0	1	2	3	4	5	ĐIỂM
Ý TƯỞNG	1. Mô tả rõ ràng, đầy đủ chi tiết theo mẫu							
	2. Ý tưởng sáng tạo, ý nghĩa, có tính thử thách và hợp logic							
	3. Cốt truyện/ chủ đề của hoạt hình							
THIẾT KẾ	4. Bối cảnh thể hiện được sự sáng tạo và trí tưởng tượng							
	5. Bối cảnh được thiết kế phù hợp với thể loại hoạt hình							
	6. Hoạt hình chứa các đối tượng có thể tạo ra và sử dụng chúng hiệu quả							
	7. Có một số lượng thích hợp các đối tượng phù hợp với loại hoạt hình							
	8. Thiết kế phối cảnh, màu sắc và các vật thể hỗ trợ cho việc chơi hoạt hình							
	9. Góc quay, hệ thống điều khiển, hiệu ứng, âm thanh, ...							
	10. Lập trình cho tất cả các đối tượng và nhân vật trong hoạt hình							
LẬP TRÌNH	11. Sử dụng một số bộ tính giờ (timer)							
	12. Sử dụng tính ngẫu nhiên, và có nhiều hơn một bảng điểm (score)							
	13. Các luật chơi được viết rõ ràng, mọi người tham gia dễ dàng							
TIẾP CẬN - XỬ LÝ	14. Có độ thử thách - người chơi cần một số nỗ lực để hoàn thành/ chiến thắng							
	15. Hoạt động trôi chảy - không có trực trặc hoặc lỗi trong khi chơi							
	16. Dễ chơi, tường thuật tốt							
ĐÁNH GIÁ	17. Thú vị/ có tính giải trí/ thu hút							
	18. Mang tính giáo dục							
	19. Phát triển thành công ý tưởng sản phẩm ban đầu							
	20. Có hướng phát triển							