はじめてのプログラミング

メンター用資料

この資料は、株式会社 Innovation Power (代表:宮島衣瑛)がScratchを用いた初心者向けワー クショップを実施する際に行っているファシリテートの中身を記録したものです。タイムライン形 式になっており、どのタイミングでどのような発問を子どもたちに投げかけるのかなどが記載さ れています。適宜参考にしながらワークショップを進めてください。

最終更新日:2018年8月14日

■ 前提条件

ワークショップは2時間想定です。参加者はコンピュータの操作は可能だが、プログラミングをし たことはない初心者を想定しています。

- ・使用ツール: Scratch 2.0 オンライン版(オフラインでも可)
- ・想定人数:20名前後
- ・年齢層:小学校3年生~中学3年生

■ ワークショップ前の準備

【ファシリテータ(進行を進める人)】

①:Scratchの教員用アカウントを取得する

Scratch の教員用アカウントは、自分のクラスを作る機能やクラスに追加する生徒のアカウント を一括で追加できる機能が実装されています。ワークショップのファシリテータをやる方にはマス トな機能ばかりですので、作成してください。

作成方法は、 次の公式サイトを参考にしてください。 https://scratch.mit.edu/educators/#teacher-accounts

※申請にはMITチームへのリクエストが必要となるため、少し時間がかかる場合があります。余裕 を持って申請を行うようにしてください。

②:クラスに追加するためのアカウント

Scratchの教員用アカウントには、生徒アカウントを一括で追加するための機能があります。次の 手順で行ってください。 ・<u>https://scratch.mit.edu/educators/classes/</u> にアクセスして、自分のクラスを作る。(すで に作成済みの場合はやらなくてよい)

・「生徒」リンクをクリックして、「複数の生徒を追加する」の「CSVをアップロード」を確認 する。



userName1	password1
userName2	password2
userName3	password3
userName4	password4
userName5	password5

csvファイルはこのような形式で用意しますが、 作るのは大変なのでジェネレーターを使用し ます。

・<u>http://kirie.me/apps/scratch-csv/</u>にアクセスして、CSVファイルを作成する。



Number of create new accounts. => 作成するアカウントの総数

Same username => 作成するユーザ名の共通部

Same password => 作成するパスワードの共通部

例)

Same username = innovation_ Same password = power_

innovation_01	power_01
innovation_02	power_02

- ・生徒アカウントCSVをアップロードする
- ・参加者にユーザ名とパスワードを教える

※このジェネレータで作成したアカウントの連番は、数字が一桁の場合は 01,02 のように先頭 に0が付きます。

今回は以下のようなプログラムをあらかじめ作っておきます。(本編の内容通り作成したもの) 猫のスクリプト



ネズミのスクリプト



Goalのスクリプト(なし)



ステージ(背景:track)



■ ワークショップの流れ

時間	内容	備考
0:01	・ ワークショップスタート! □ファシリテータ挨拶 □メンター紹介 □本日のテーマなどの発表	
	// もし、ほとんどの子が初めて参加する場合や入場時に 静かすぎた場合には、アイスブレイクを行う。	
	/*	
	・アイスブレイク 目的:ワークショップ参加者の関係性を柔らかくする。 アイスブレイクの手法はたくさんあるが、ここではいくつ か紹介しておく。	
	 ③:拍手ゲーム 円上になり、始点の人を決める。始点の人は合図が鳴ったら、手を叩く。その隣にいる人は始点の人が手を叩いたら手を叩き、その隣にいる人も…とつなげていきなるべく早く、最後の人まで拍手が繋げるようにするゲーム途中で ・どうすれば早くなるか隣の人と考えてみよう ・どうやったら早くなるか、意見がある人は手をあげてください。など発言できる場を設ける。 	早くなるには ・最初の掛け声を決める ・隣同士はなるべく近くに
	何回かやってみる。最後に拍手をして終わる。	
	*/	

	ワークショップ本編:フライ!		
00:10	Section 1 (20分)		
	今回作るゲームの紹介		
	・今回は、猫が飛びながら障害物をよけゴールまで目指 すゲームを作成する。		
	・具体的にはこのようなゲームである。(あらかじめ作成 したものを実行)		
	・最初は猫の動きをプログラムしていく		
	プログラムの準備 ・ブラウザを開かせる Google Chrome または Microsoft Edge を推奨 Internet Explorer 以外であればなんでもよい		
	・ <u>https://scratch.mit.edu</u> にアクセス "Scratch"とGoogleで検索してもよい。		
	 Scratchの説明をする(初回の人が多い場合) Scratchはアメリカのマサチューセッツ工科大学メディアラボが開発した教育用のプログラミングツールである 全世界で使われており、いま世界で最も有名なプログラミングツールのひとつ 		
	さっそくやってみる ・画面左上の「作る」ボタンを押して、エディタ画面に飛 ぶ。画面の説明をする。		



猫を上下左右に動かす	
 ・猫は上下左右の矢印キーを押すとそれぞれの方向へ動く。この動きはどんなブロックを使うと作ることができるのか考えてみよう。 答え: 【制御】の<もし>と【調べる】の<スペースキーが押された>ブロックを使う 	ヒント 【制御】にあるブロックと【調べ る】にあるブロックを使って作る よ。 「もしキーが押されたなら」とい うプログラムを作から…
・早速プログラミングしていこう。	
まず【イベント】からく旗がクリックされたとき>ブロッ クを右側に持ってくる	
その下に【制御】から<ずっと>を右側に持ってきて、< 旗がクリックされたとき>ブロックの下にくっつける。	
【制御】から くもし> を右側に持ってきて、 くずっと> ブ ロックの下にくっつける。	
<mark><もし></mark> ブロックの穴に【調べる】から<スペースキーが 押された>ブロックを持ってきて入れる	
<pre></pre>	
 がクリックされたとき すっと もし スペース キーが押された なら スペース 上向き矢印 古向き矢印 左向き矢印 どれかの a b 	
ここから、上向き矢印キーが押されたときの動きをプロ グラムしていく。	
【動き】から <y座標を●ずつ変える>を右側に持ってき てくもし上向き矢印キーが押された<mark>なら</mark>>ブロックの下 にくっつける。</y座標を●ずつ変える>	









・まず、右側から左に障害物を移動させるプログラムを考	ヒント
えて作ってみよう。	基本の<旗がクリックさらた時>
	くずっと>を使う
答え:「ずっと、x座標を-5ずつ変える」というプログラ	「左向き矢印キー」の時に使った
ムを作成する	プログラムを参考にしよう
・早速作成する	
<旗がクリックさらた時>を右側に持ってきたあと、<ず	
<mark>っと></mark> をその下につける。	
<mark><ずっと></mark> の中に <x座標を●ずつ変える>ブロックを入</x座標を●ずつ変える>	
れ数字を「10」から「-3」に変える	
実行すると、障害物が左に移動していることがわかる。	
・しかし、このままだと障害物が左端についても右側に	
戻ってこない。これを変えるプログラムを作っていく	
<mark><もし></mark> ブロックを右側に持ってきて、 <mark><ずっと></mark> の中に	
入れる	
· · · · · · · · · · · · · · · · · · ·	
【演算】から<●<●>(小なりブロック)を右側に持	
ってくる。	
【動き】からくν咴堙>ブロックを右側に持ってきてく●	
<●>ブロックの左側の穴に入れる。	
×座標)	
<●<●>フロックの右側の穴に '-240」と入力する	
<x座標<-240>ブロックをブロックの穴に入れる。</x座標<-240>	
このブロックは「x座標が-240よりも小さいとき」という 意味になる。	
・ステージの左端は-240だから	scratchには「端に触わたら」と
左端に着いた時という意味になる	いうブロックもありますが、この
	ブロックを使うと、上下の端に触し
がクリックされたとき	れた時も右側に戻ってしまいま
ずっと	す。そのため、ここでは
x座標を 🕒 ずつ変える	「x<240」ブロックを使って端に
もし ×座標 < -240 なら	着いた時を表現しています。

・次は、もし左端に着いた時「右側に戻す」というプロ グラムを作成しよう。	۲J
【動き】から <x座標を●、y座標を●にする>ブロック を右側に持ってきて<もし<x座標 -240="" <="">なら>ブロ ックの中に入れる。</x座標></x座標を●、y座標を●にする>	
<x座標を●、y座標を●にする>のブロックのx座標の数 字を「250」にして実行する</x座標を●、y座標を●にする>	
実行すると左端に着いたら、右側に移動していることが わかる。	
がクリックされたとき ずっと X座標を -3 ずつ変える もし (X座標) < -240 なら X座標を 250 、Y座標を 101 にする	
・次に右側から出てくる時に、高さがランダムになるよう にプログラムを改良する	
・どうすれば、高さがランダムになるのか考えてみよう。	ヒント 【演算】の中のブロックを使う
答え 【演算】の<●から●までの乱数>を使う	
・乱数とはランダムな数字のことである。これを使うと位 置をランダムに変えることができる。	
【演算】から<●から●までの乱数>を右側に持ってき て <x座標を●、y座標を●にする>ブロックの右側の穴 に入れる。</x座標を●、y座標を●にする>	
<●から●までの乱数>の左側の数字を「-240」に、右 側の数字を「240」にする。	
実行すると、高さがランダムになっていることがわかる	
がクリックされたとき ずっと X座標を 3 ずつ変える もし X座標 < -240 なら X座標を 230 、Y座標を -240 から 240 までの乱数 にする	











スプライト 新しいスプライト:	康士を見なから、2,3人できてき たらファシリテータも説明を始め る
クリックした後、スプライトの名前を入力する欄がある ので、ここに「Goal」と入力する	
 Goal x: 129 y: -7 向き: 90° 回転の種類: ○ ↔ ● ブレイヤーでドラッグできるか: ■ 表示する: ▼ 	
・次に「隠れて、30秒に黄色い線が右側に現れる」とい うことをプログラミングしていく。このプログラムを自分 で作ってみよう。	
まずく旗がクリックされた時>を右側に持ってきて、 【見た目】からく隠れる>をく旗がクリックされた時> にくっつける。	
次に、【制御】からく●秒待つ>ブロックを右側に持っ てきて<隠れる>の下にくっつける。	
<●秒待つ>の数字を「1」から「30」に変える。	
【動き】から <x座標●、y座標●にする>ブロックを右 側に持ってきて<mark><30秒待つ></mark>ブロックの下にくっつける</x座標●、y座標●にする>	ヒント
<x座標●、y座標●にする>のx座標の数字を「250」 に、y座標の数字を「0」にする</x座標●、y座標●にする>	障害物が左に動くプログラムを参 考にしよう
最後に【見た目】から<表示する>ブロックをブロック の下にくっつける	
がクリックされたとき 聴す 30 秒待つ x座標を 250 、y座標を ① にする 表示する	
・これで実行しても、250は右端よりも右側なので黄色い 線を見ることができない。	

・そ	·のため次の「黄色い線が左側に移動する」のプログ ·を作った後に確認する	
・黄 みよ	ē色い線を左側に移動するプログラムを自分で作って ∶う	
答え 「ず ば良	- [*] っとx座標を-5ずつ変える」というプログラムを作れ !い	
く ず ック	「っと>ブロックを右側に持ってきて<表示する>ブロ 」の下にくっつける	
<x ック</x 	座標●ずつ変える>ブロックを右側に持ってきてブロ 'の<ずっと>の中に入れる	
< x/	座標●ずつ変える>ブロックの数字を「10」から 5」に変える 「 ^{#//リックされたとま} 	ヒント 「●に触れた」というのは【調べ る】にある
・こ	れを実行すると30秒後に現れ、左側に移動すること かる	
	► A K: -240 ¥: 110 @	答え: くもし>ブロックと【調べる】の くマウスポインターに触れた>ブ
・次 線に 成す	マにスプライトから猫のスプライトを選択し「黄色い 触れるとゲームクリアになる」というプログラムを作 る	ロックを使う · <もし>ブロックを右側に持っ
まず っと た時	[*] <旗がクリックされた時>を右側に持ってきて、<ず >ブロックを右側に持ってきて<旗がクリックされ ▶>ブロックの下にくっつける	てきての くすっと> の中 ・【調べる】からくマウスポイン ターに触れた>ブロックを持って きて <mark>くもし</mark> >ブロックの穴に入れ
・「 はど	もしGOALに触れたなら」というプログラムを作るに のブロックを使えばいいか考えてみよう	3







01:21	Section4(35分)		
	改造タイム ・改造して、自分だけのオリジナルかくれんぼゲームを作ろう (改造案) ・クリア時に猫以外を隠す ・障害物に当たった数をカウントし、ゲームオーバー画面を作る ・障害物や背景を追加する 	子供達には、改造タイムの残り時 間を必ず5・10分おきくらいに伝 えてあげてください。	
	鑑賞&発表タイム ・最後に他の人の作品で遊んでみよう		
	・他の人と自分の作品の違いを考えてみよう		
01:55	・時間があれば、発表を行う		
	終わり		
01:56 02:00	ワークショップ終わりの挨拶をする。		