



Guia PCROM: Resoluções Scratch

Na Tabela 1 seguem as questões que serão aplicadas aos conceitos apresentados com Scratch e a indicação do conteúdo a ser abordado por cada uma delas. O enunciado, os conceitos necessários e a resolução dos problemas podem ser encontrados nos anexos indicados:

Tabela 1 - Identificação das questões de olimpíadas no Scratch

Questão	Categoria do PIC	Conteúdo	Fonte	ANEXO
1	Áreas	1º ano do Ensino Médio	OBMEP 2012 - Nível 1 (Fase 2)	A
2	Métodos de Contagem e Probabilidade	3º ano do Ensino Médio	Questão 22 do Canguru Matemático sem Fronteiras - categoria Escolar (2017)	B
3	Métodos de Contagem e Probabilidade	9º ano do Ensino Fundamental	Construindo Pipas" para o Ensino Fundamental - Disponível no Portal do Saber – OBMEP (Adaptação da questão Colorindo um Mapa, do Banco de Questões da OBMEP, ano 2010)	C
4	Iniciação à aritmética	1º ano do Ensino Médio	Questão 19 da OBMEP 2019 (Nível 2)	D
5	Grafos - Uma introdução	2º ano do Ensino Médio	Quebra-cabeças de Matemática para o Ensino Fundamental - Portal do Saber - OBMEP	E

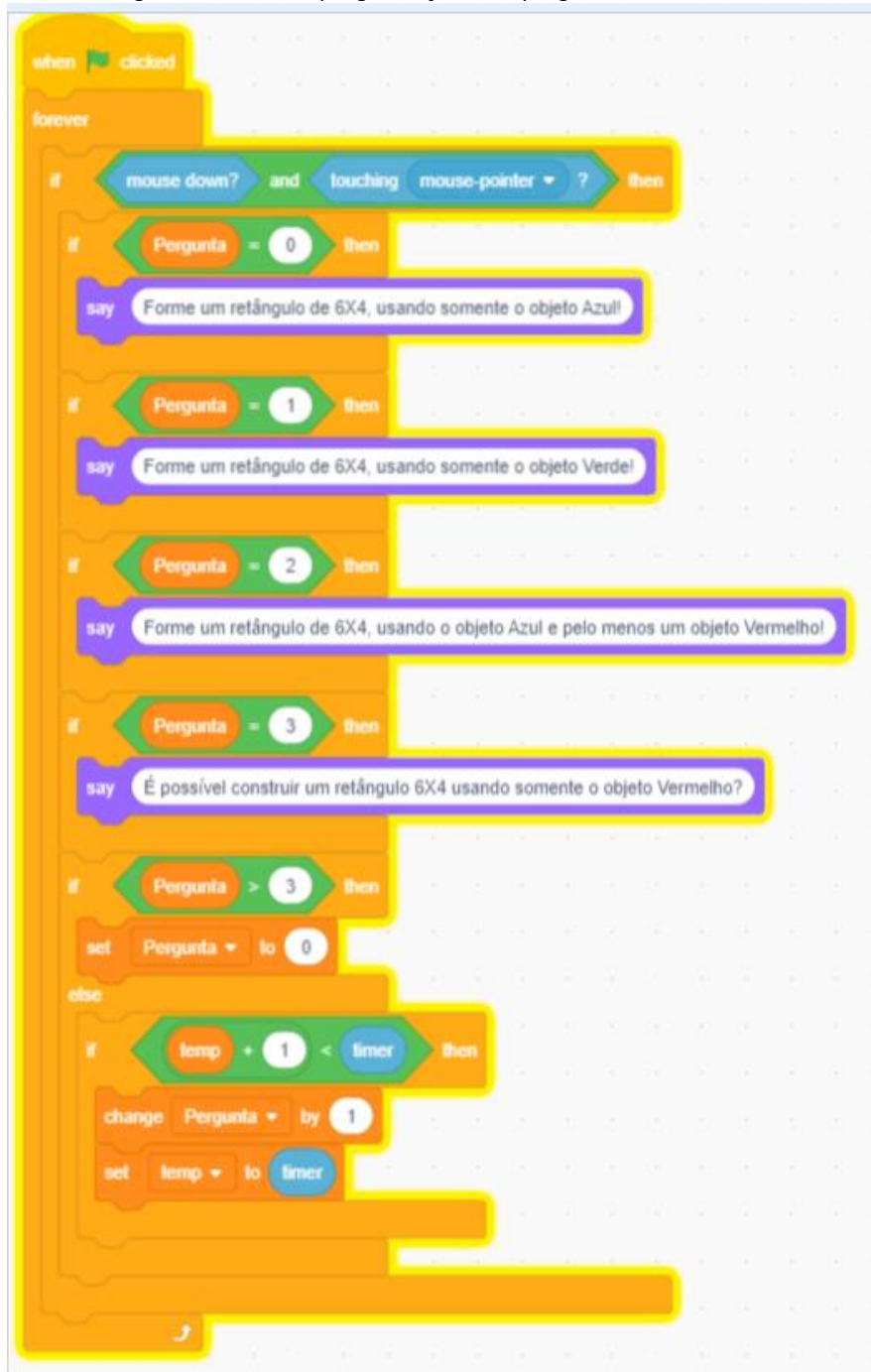
Fonte: Autora

APÊNDICE B – Solução Scratch (Questão 1 - TETRIS)

Explique que depois de realizada a identificação, o próximo passo é a construção para os blocos A, B e C, onde devem ser realizados movimentos de rotação e translação para encaixe das peças.

Depois é realizada a programação do bloco através da inserção de perguntas (Figura 1) que aparecem na tela com o objetivo de guiar a construção da resolução de acordo com o que está sendo pedido em cada um dos blocos (A, B e C).

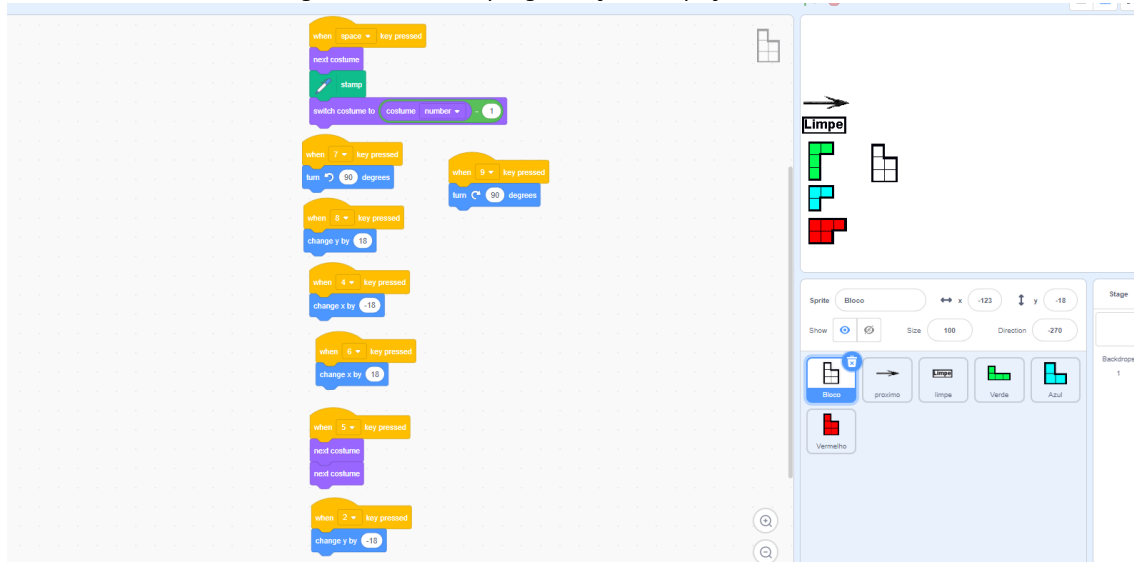
Figura 1 - Bloco de programação com perguntas estruturadas



Fonte: Autora

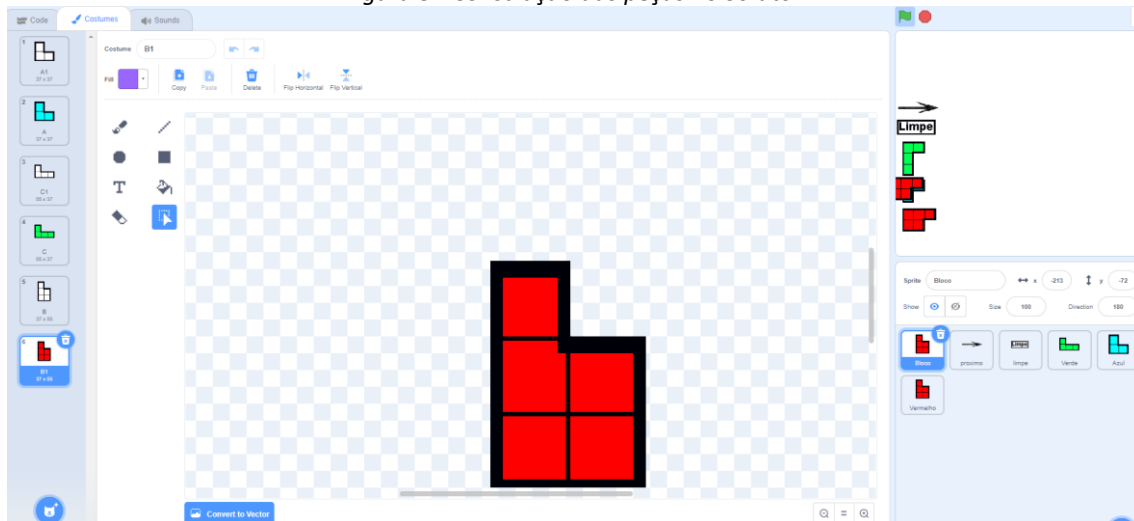
Então, é realizada a programação em cada um dos objetos construídos que serão utilizados para realizar a rotação e translação dos objetos (Figura 2), ou seja, o movimento necessário para encaixe das peças. Além disso, é configurado o bloco A com a cor azul, o B com a cor verde e o C com a cor vermelha (Figura 3).

Figura 2 - Bloco de programação das peças no Scratch



Fonte: Autora

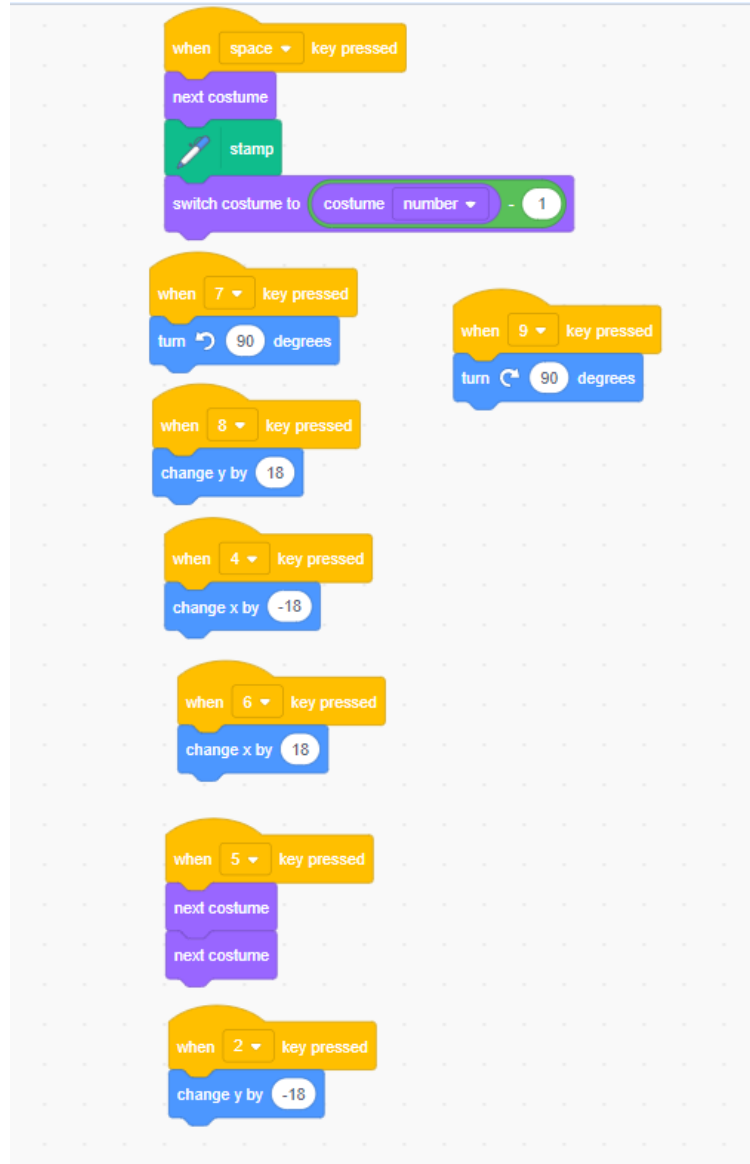
Figura 3 - Construção das peças no Scratch



Fonte: Autora

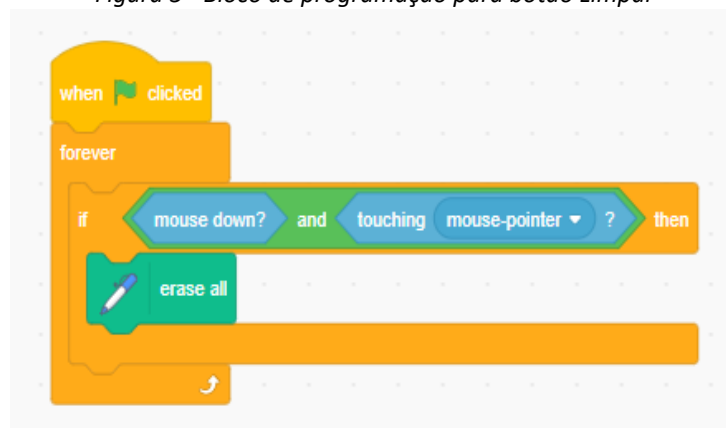
Quanto às peças, foi construído um bloco de instruções para possibilitar o deslocamento tanto para a direita e esquerda, quanto para cima e para baixo. Ou seja, é possível rotacionar para uma dessas direções ou até mesmo mudar de peça através dos seguintes comandos que são acionados utilizando as teclas (7-rotacionar 90° para a esquerda; 8-para cima; 4-esquerda; 6-direita; 9-rotacionar 90° para a direita; 2-para baixo; 5-alterar tipo de peça) e que podem ser visualizados na Figura 4.

Figura 4 - Bloco de programação para movimentação das peças



Fonte: Autora

Figura 5 - Bloco de programação para botão Limpar

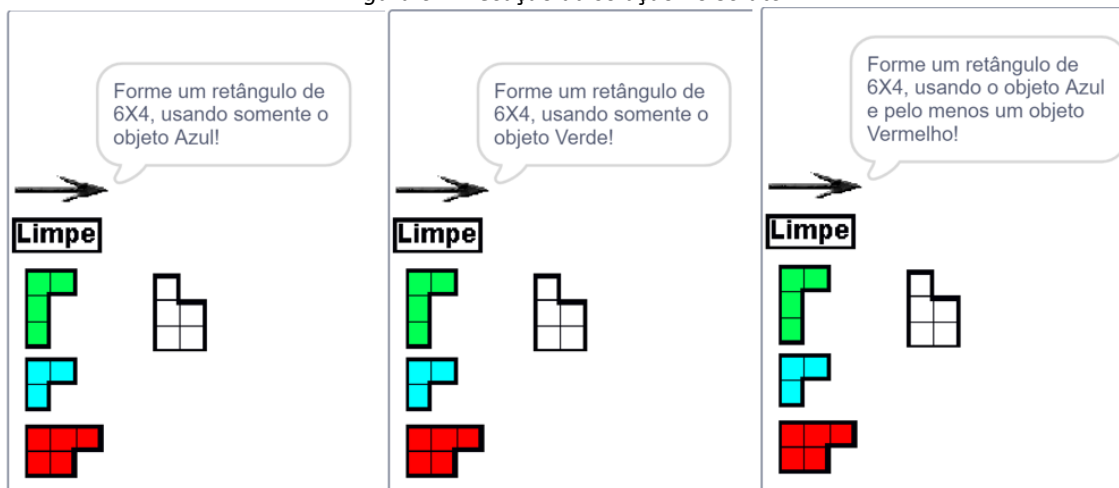


Fonte: Autora

Durante o processo de resolução do problema e construção da solução foi necessário identificar os tipos de formas geométricas disponíveis e construir um raciocínio baseado na rotação de translação dessas figuras geométricas.

Após construção da resolução no *Scratch*, é possível visualizar de acordo com a Figura 6 a resolução em execução, sendo guiada pelas perguntas (a cada clique na seta) e dando sequência a construção das soluções por meio do encaixe das formas geométricas, atuando como auxílio para que o aluno possa visualizar as figuras, calcular a área e chegar a solução do problema.

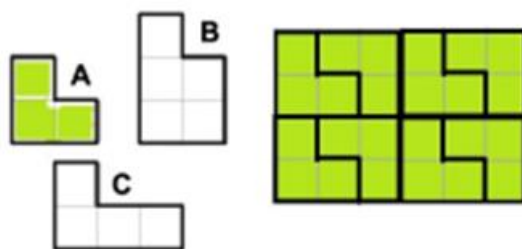
Figura 6 - Execução da solução no Scratch



Fonte: Autora

Explique a resolução para o item A mostrando a ligação entre a solução matemática e a execução do problema:

Figura 7 - Resolução matemática do item A



Fonte: OBMEP

Figura 8 - Resolução no Scratch do item A

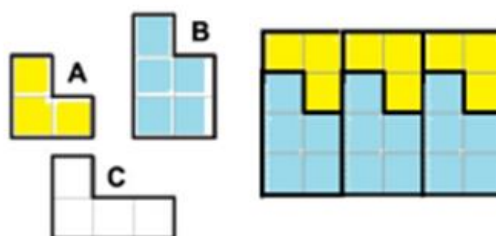


Fonte: Autora

Explique a resolução para o item B mostrando a ligação entre a solução matemática e a execução do problema:

É necessário visualizar que uma peça do tipo B é o equivalente a 5 quadradinhos, e nesse caso, as peças não devem ser sobrepostas, portanto, 2 peças do tipo B cobrem 10 quadradinhos, 3 peças cobrem 15, 4 peças cobrem 20 e 5 peças cobrem 25, ou seja, passa de 24 quadradinhos. Por isso, é possível visualizar que não pode ser coberto apenas com peças do tipo B.

Figura 9 - Resolução matemática do item B



Fonte: OBMEP

Explique a resolução para o item C mostrando a ligação entre a solução matemática e a execução do problema:

Por meio da visualização do bloco vermelho é possível verificar que não é possível resultar em um retângulo de 6x4. Isso acontece porque é exigido que as peças não sejam colocadas em sobreposição e os quadradinhos cobertos por blocos do tipo vermelho são resultantes de 5, porém, o retângulo solicitado possui uma área de 24 (não múltiplo de 5).

Essa verificação pode ser realizada da seguinte forma: aplicando o conceito de divisibilidade, ou seja, dividindo o valor da área pela possibilidade de cobertura da peça proposta e assim verificando se um é múltiplo do outro. Através desse raciocínio é possível obter a resposta final. Portanto, nesse tipo de exercício o

aluno coloca em prática conceitos além de área explorando diferentes contextos com formas geométricas e trazendo conceitos como rotação, translação, possibilidade e divisibilidade.

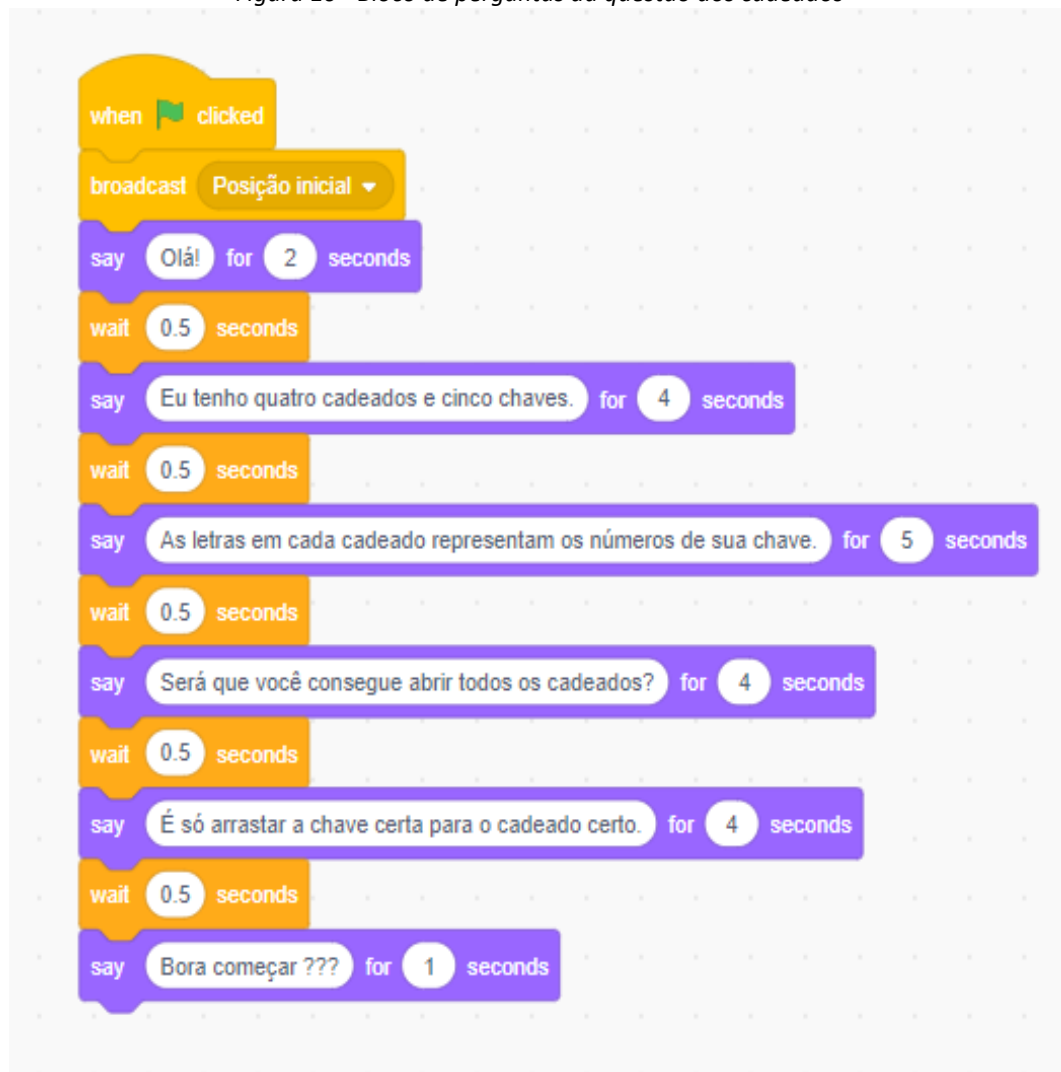
A solução completa no *Scratch* pode ser acessada através do seguinte link:
<https://scratch.mit.edu/projects/14550632/>

Nesse passo é esperado que os alunos questionem as etapas de montagem do algoritmo disponibilizado, tanto quanto as cláusulas estruturais quanto ao raciocínio utilizado para a solução. E para cada questionamento, o professor deve realizar explicação relacionando cada linha de código com situações práticas para que a implementação torne-se cada vez mais parte das soluções que forem desenvolver para os mais diversos problemas que compreendem situações do cotidiano.

APÊNDICE D – Solução Scratch (Questão 2 - CADEADOS)

O primeiro bloco é estruturado através das perguntas que são como gatilhos para o início das ações. Ou seja, As perguntas conforme descritas na Figura 10, são uma sequência de frases que guiam o aluno a entender o problema mencionado.

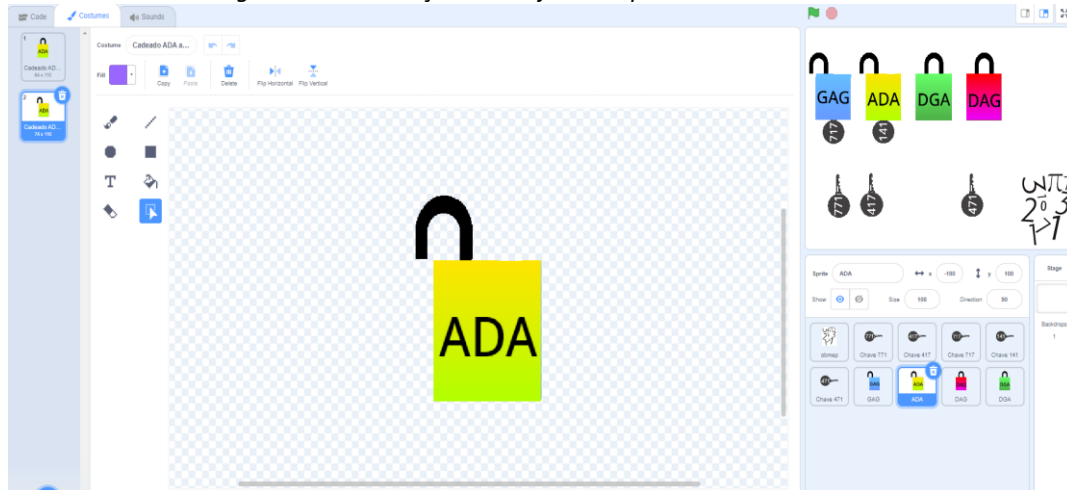
Figura 10 - Bloco de perguntas da questão dos cadeados



Fonte: Autora.

Após guia de como o problema deve ser conduzido, na Figura 11 é possível visualizar a construção dos elementos visuais, sendo assim, a construção de cada um dos cadeados e das chaves.

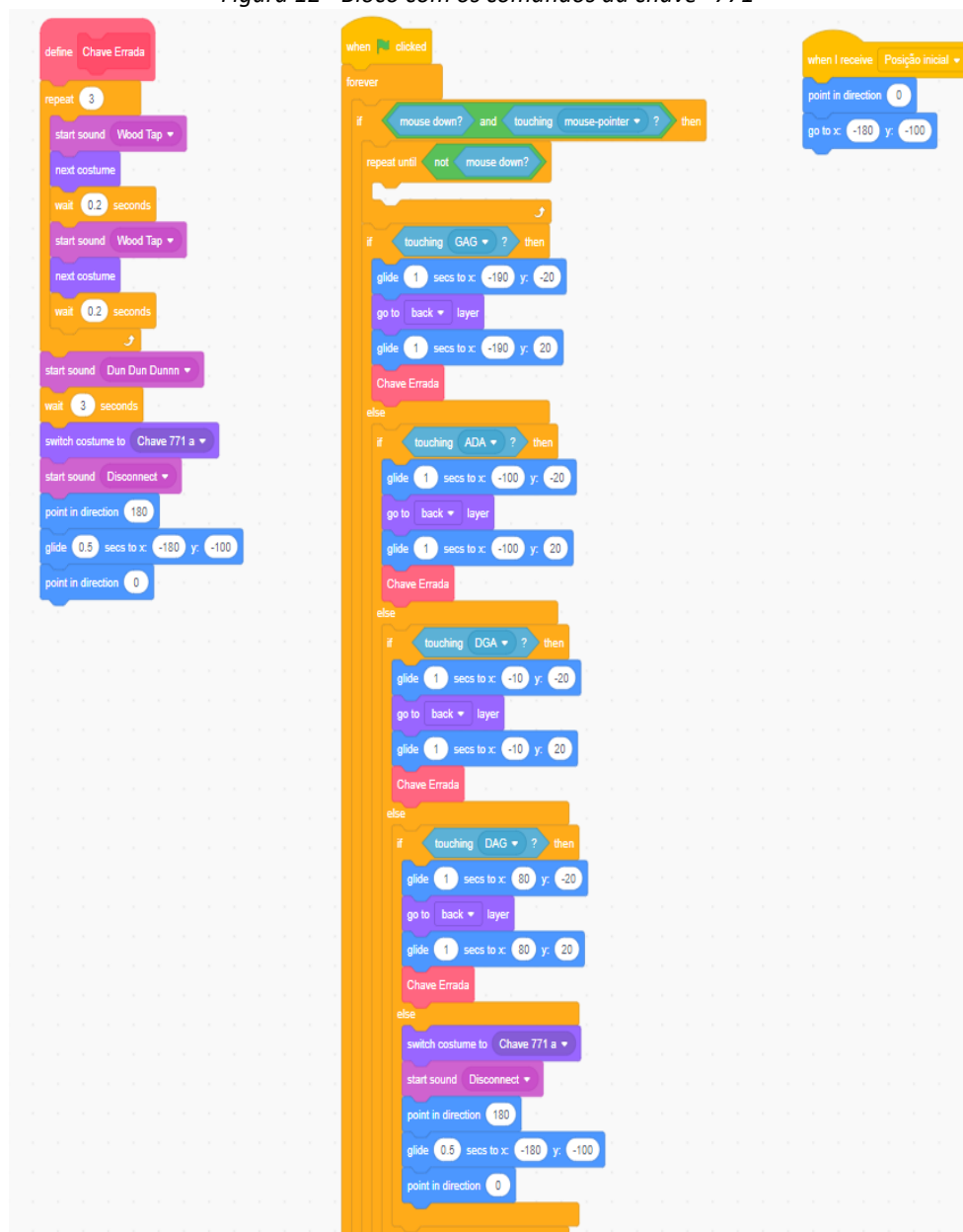
Figura 11 - Construção dos objetos do problema do cadeado



Fonte: Autora

A cada uma das chaves é atrelado um bloco de código que possibilita a realização das ações para que o problema seja resolvido. Portanto, na Figura 12 são demonstrados os blocos correspondentes a chave “771”. No primeiro bloco, que podemos definir por “Chave Errada” é definido um conjunto de ações caso a chave não esteja no cadeado correspondente. Essa verificação é realizada através do posicionamento da chave (ao final do bloco pelas variáveis x e y). O segundo bloco trata-se da verificação de cada um dos cadeados, que podem ser identificados por “GAG”, “ADA”, “DGA” e “DAG”(sendo a última possibilidade, identificado na cláusula “else” desse bloco). Assim como as chaves, os blocos também são localizados através de seus posicionamentos (identificados pelas variáveis “x” e “y”). É importante notar que, no segundo bloco é sempre realizada a chamada do primeiro bloco, que é referente a verificação da “Chave errada”. E após isso temos o terceiro bloco que retorna a chave para a posição inicial após realizar a tentativa de abertura do cadeado.

Figura 12 - Bloco com os comandos da chave “771”

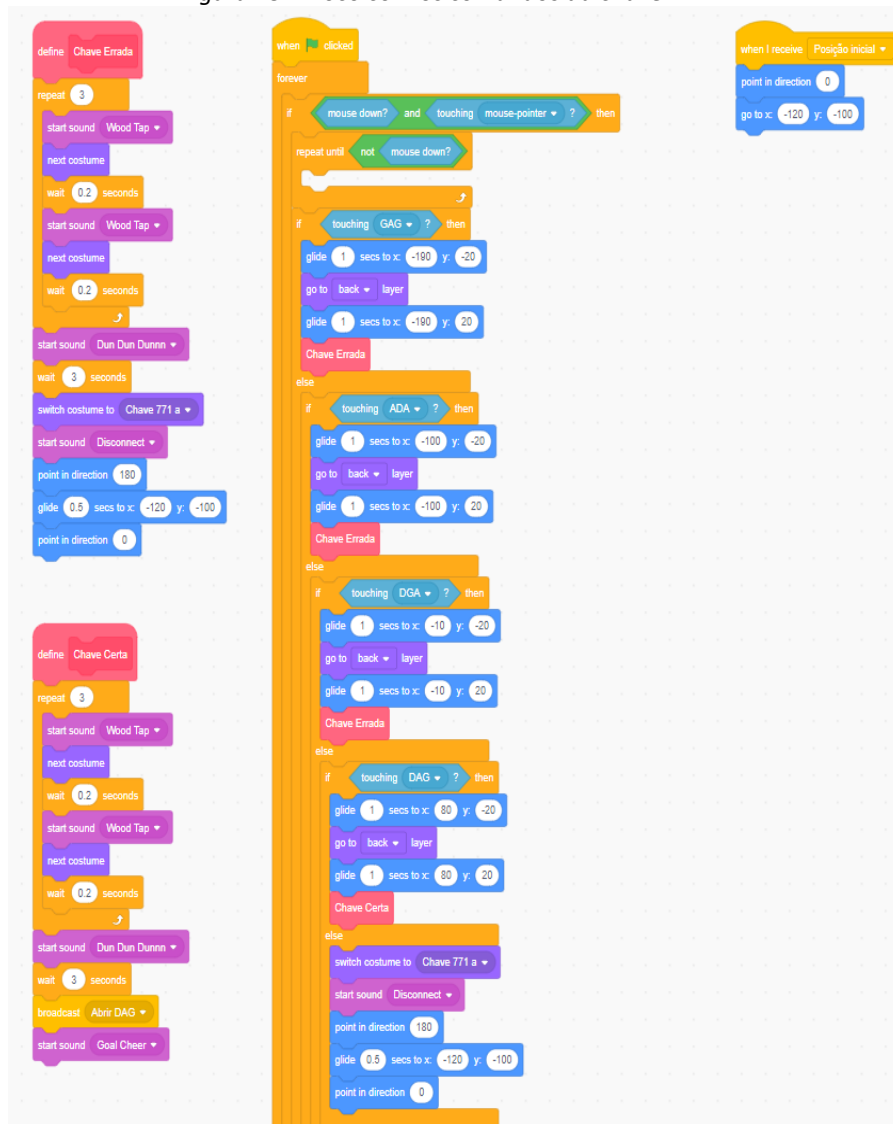


Fonte: Autora

Como mencionado anteriormente, a cada uma das chaves é atrelado um bloco de código que possibilita a realização das ações para que o problema seja resolvido. Portanto, na Figura 13 são demonstrados os blocos correspondentes a chave “417”. No primeiro bloco, que podemos definir por “Chave Errada” é definido um conjunto de ações caso a chave não esteja no cadeado correspondente. Essa verificação é realizada através do posicionamento da chave (ao final do bloco pelas variáveis x e y). Além disso, temos um bloco abaixo que é correspondente a “Chave Certa”, onde caso essa chave seja colocada no cadeado “DAG” o mesmo deve ser aberto, e essa verificação é realizada através da última cláusula do segundo bloco. Esse segundo bloco trata-se da verificação de cada um dos cadeados, que podem ser identificados por “GAG”, “ADA”, “DGA” e “DAG” (sendo a última possibilidade, contendo a verificação para a chave certa conforme já mencionado). Assim como as chaves, os blocos também são localizados através de seus posicionamentos (identificados pelas variáveis “x” e “y”). É importante

notar que, nesse segundo bloco é sempre realizada a chamada do primeiro bloco, que é referente a verificação da “Chave errada”, com exceção da “Chave Certa”. E após isso temos o terceiro bloco que retorna a chave para a posição inicial após realizar a tentativa de abertura do cadeado.

Figura 13 - Bloco com os comandos da chave “417”



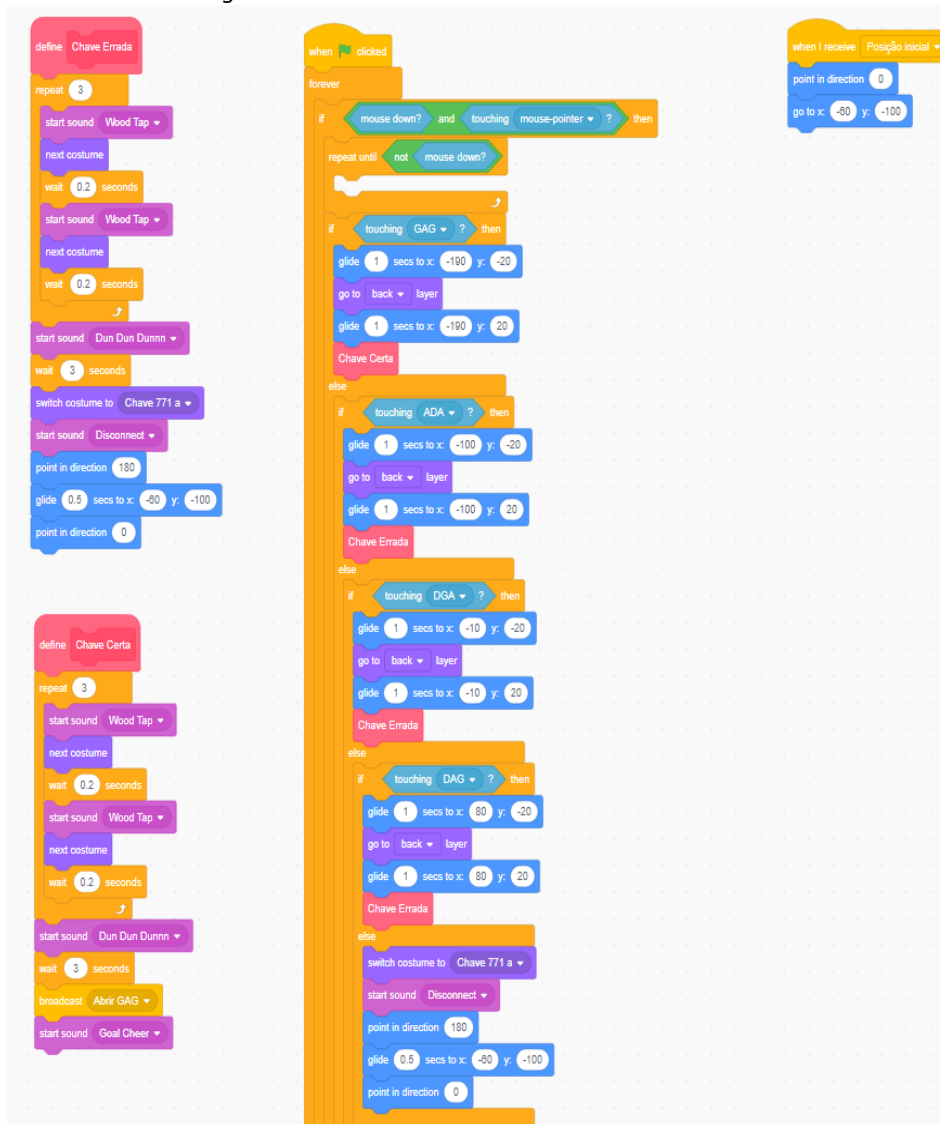
Fonte: Autora

Ainda seguindo o mesmo raciocínio, a cada uma das chaves é atrelado um bloco de código que possibilita a realização das ações para que o problema seja resolvido. Portanto, na Figura 14 são demonstrados os blocos correspondentes a chave “717”. No primeiro bloco, que podemos definir por “Chave Errada” é definido um conjunto de ações caso a chave não esteja no cadeado correspondente. Essa verificação é realizada através do posicionamento da chave (ao final do bloco pelas variáveis x e y). Além disso, temos um bloco abaixo que é correspondente a “Chave Certa”, onde caso essa chave seja colocada no cadeado “ADA” o mesmo deve ser aberto, e essa verificação é realizada através da última cláusula do segundo bloco.

Esse segundo bloco trata-se da verificação de cada um dos cadeados, que podem ser identificados por “GAG”, “ADA”, “DGA” e “DAG” (sendo a segunda possibilidade, contendo a verificação para a chave certa conforme já mencionado). Assim como as chaves, os blocos também são localizados através

de seus posicionamentos (identificados pelas variáveis “x” e “y”). É importante notar que, nesse segundo bloco é sempre realizada a chamada do primeiro bloco, que é referente a verificação da “Chave errada”, com exceção da “Chave Certa”. E após isso temos o terceiro bloco que retorna a chave para a posição inicial após realizar a tentativa de abertura do cadeado.

Figura 14 - Bloco com os comandos da chave “717”

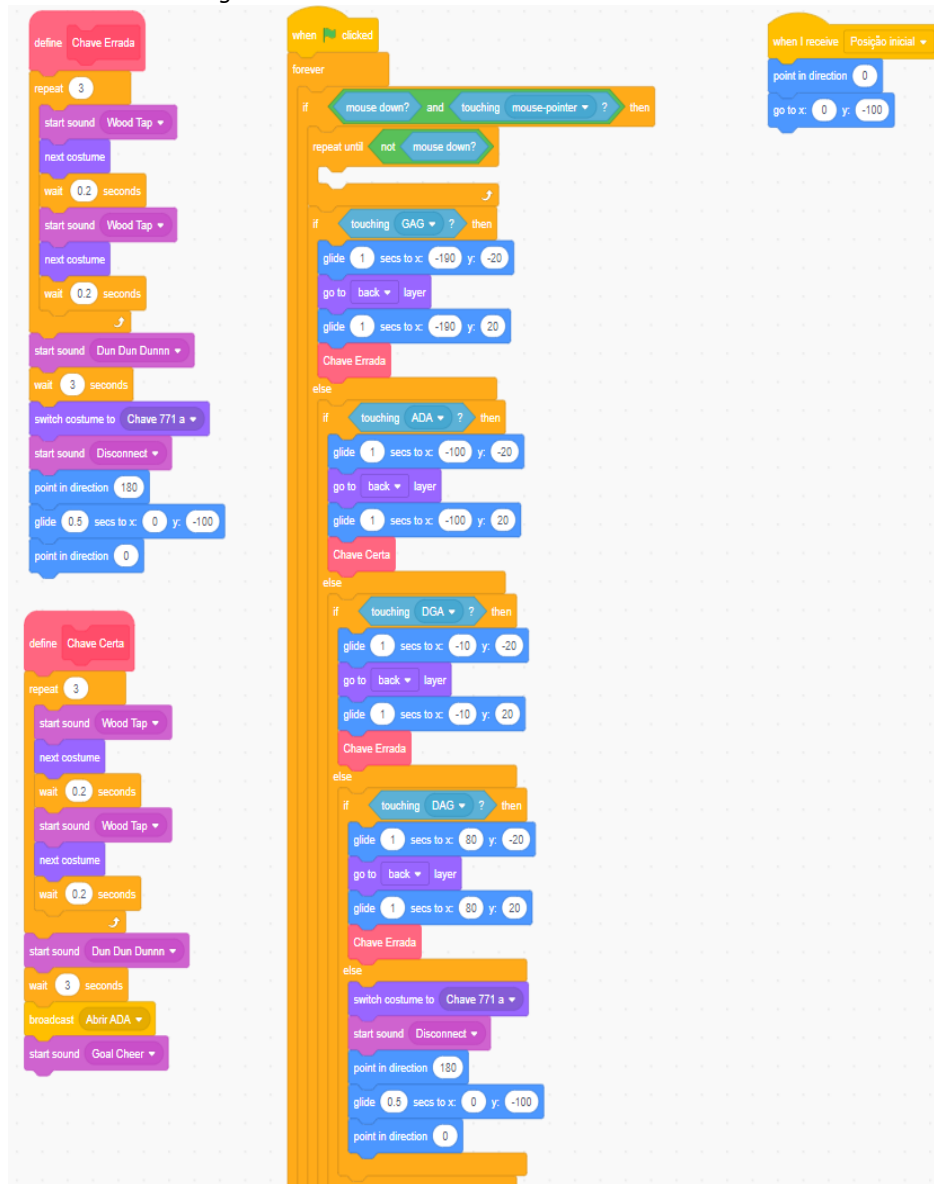


Fonte: Autora

Para finalizar a lógica atrelada às chaves, na Figura 15 são demonstrados os blocos correspondentes a chave “141”. No primeiro bloco, que podemos definir por “Chave Errada” é definido um conjunto de ações caso a chave não esteja no cadeado correspondente. Essa verificação é realizada através do posicionamento da chave (ao final do bloco pelas variáveis x e y). Além disso, temos um bloco abaixo que é correspondente a “Chave Certa”, onde caso essa chave seja colocada no cadeado “DGA” o mesmo deve ser aberto, e essa verificação é realizada através da última cláusula do segundo bloco. Esse segundo bloco trata-se da verificação de cada um dos cadeados, que podem ser identificados por “GAG”, “ADA”, “DGA” e “DAG” (sendo a terceira possibilidade, contendo a verificação para a chave certa conforme já mencionado). Assim como as chaves, os blocos também são localizados através de seus posicionamentos (identificados pelas variáveis “x” e “y”). É importante

notar que, nesse segundo bloco é sempre realizada a chamada do primeiro bloco, que é referente a verificação da “Chave errada”, com exceção da “Chave Certa”. E após isso temos o terceiro bloco que retorna a chave para a posição inicial após realizar a tentativa de abertura do cadeado.

Figura 15 - Bloco com os comandos da chave “141”



Fonte: Autora

Anteriormente foi exposta a lógica utilizada para as chaves, a seguir será demonstrada a lógica utilizada para os cadeados. De acordo com a Figura 16, os dois primeiros blocos são referentes a verificação se o cadeado está fechado ou aberto. E o último bloco é para verificar se a chave que está sendo colocada é a chave correta, ou seja, a chave que abre esse cadeado, e nesse caso a chave correspondente é a “717”, para isso é chamado o bloco que já foi criado e explicado anteriormente: o bloco que diz respeito ao raciocínio utilizado para verificação do cadeado “GAG” com a chave “717”.

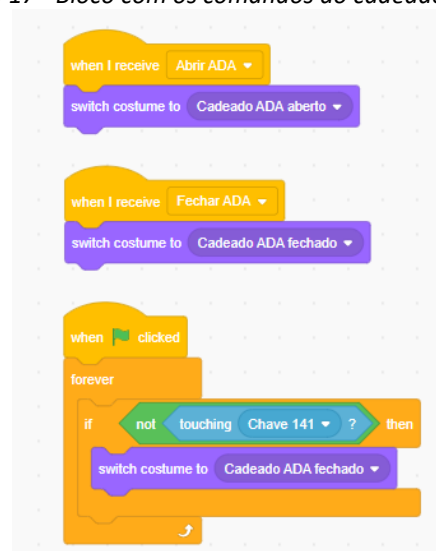
Figura 16 - Bloco com os comandos do cadeado “GAG”



Fonte: Autora

De acordo com a Figura 17, os dois primeiros blocos são referentes a verificação se o cadeado está fechado ou aberto. E o último bloco é para verificar se a chave que está sendo colocada é a chave correta, ou seja, a chave que abre esse cadeado, e nesse caso a chave correspondente é a “141”, para isso é chamado o bloco que já foi criado e explicado anteriormente: o bloco que diz respeito ao raciocínio utilizado para verificação do cadeado “ADA” com a chave “141”.

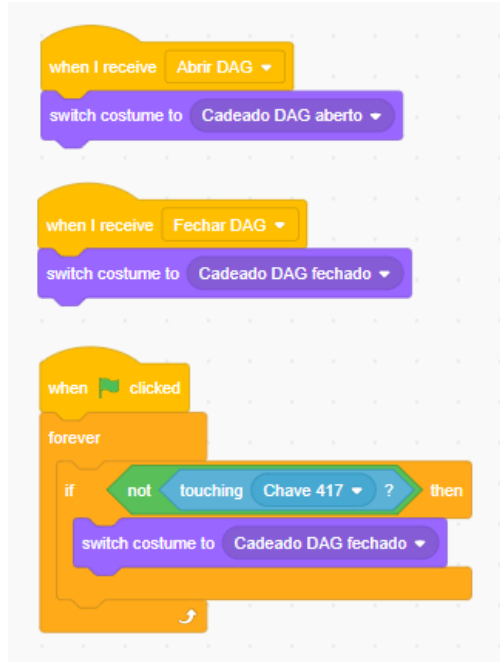
Figura 17 - Bloco com os comandos do cadeado “ADA”



Fonte: Autora

Na Figura 18 é possível verificar que os dois primeiros blocos são referentes a verificação se o cadeado está fechado ou aberto. E o último bloco é para verificar se a chave que está sendo colocada é a chave correta, ou seja, a chave que abre esse cadeado, e nesse caso a chave correspondente é a “417”, para isso é chamado o bloco que já foi criado e explicado anteriormente: o bloco que diz respeito ao raciocínio utilizado para verificação do cadeado “DAG” com a chave “417”.

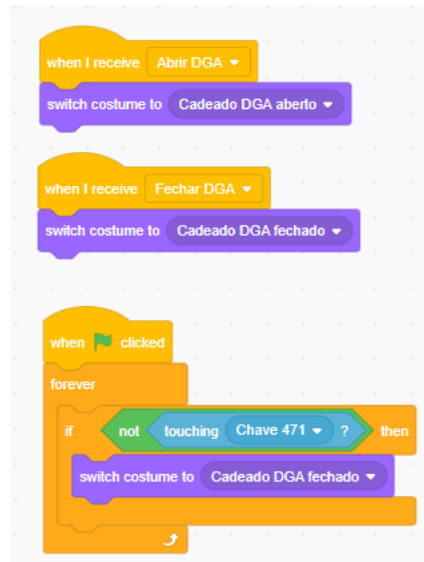
Figura 18 - Bloco com os comandos do cadeado “DAG”



Fonte: Autora

Na Figura 19 é possível verificar que os dois primeiros blocos são referentes a verificação se o cadeado está fechado ou aberto. E o último bloco é para verificar se a chave que está sendo colocada é a chave correta, ou seja, a chave que abre esse cadeado, e nesse caso a chave correspondente é a “471”, para isso é chamado o bloco que já foi criado e explicado anteriormente: o bloco que diz respeito ao raciocínio utilizado para verificação do cadeado “DGA” com a chave “471”.

Figura 19 - Bloco com os comandos do cadeado “DGA”



Fonte: Autora

Além disso, é importante destacar que a chave “771” é a única que não contém o bloco “Chave Certa” porque é a única chave que não abre nenhum dos cadeados.

Após construída a solução no Scratch, que trabalha a verificação de chaves para abertura de cadeados tendo a solução do problema através do método de

combinação e contagem é possível realizar a execução dessa solução de acordo com o ilustrado nas figuras a seguir.
De acordo com a Figura 20 podemos verificar a execução do bloco de perguntas que tem como objetivo guiar o aluno a entender o problema e posterior solução.

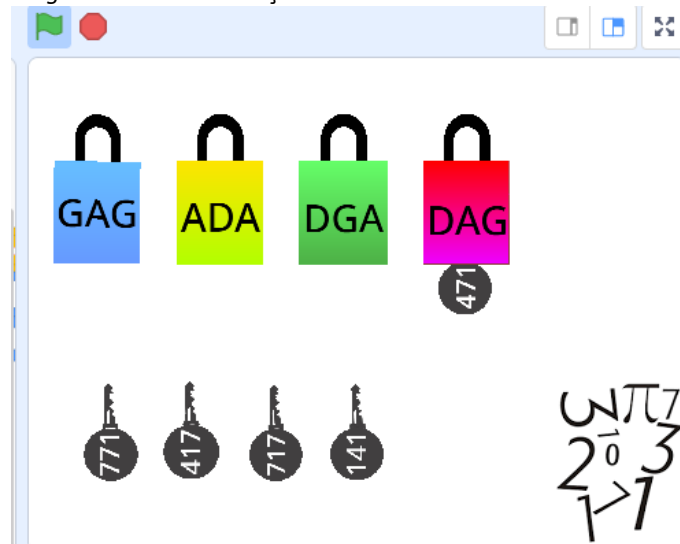
Figura 20 - Execução da solução no Scratch (perguntas)



Fonte: Autora

A Figura 21 demonstra o resultado quando é realizada uma tentativa em que a chave não corresponde àquele cadeado. Nesse caso, os blocos explicados anteriormente ("Chave certa", "Chave errada" e dos cadeados) são verificados e o cadeado não é aberto.

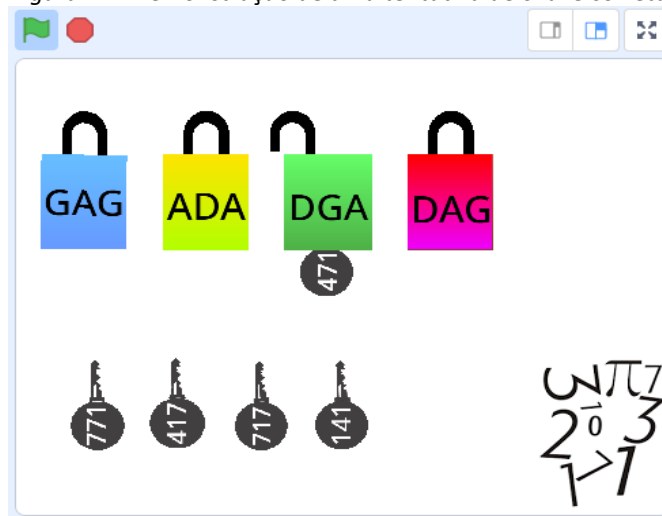
Figura 21 - Demonstração de uma tentativa de chave errada



Fonte: Autora

A Figura 22 demonstra o resultado quando é realizada uma tentativa em que a chave corresponde àquele cadeado. Nesse caso, os blocos explicados anteriormente ("Chave certa", "Chave errada" e dos cadeados) são verificados e o cadeado é aberto.

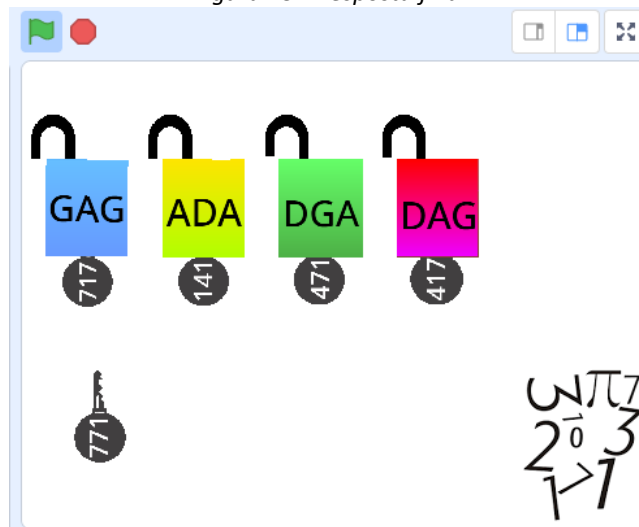
Figura 22 - Demonstração de uma tentativa de chave correta



Fonte: Autora

Após tentativas com as combinações de chaves e cadeados, pode ser visualizado na Figura 23 a resposta final do problema com todas as chaves correspondentes.

Figura 23 - Resposta final

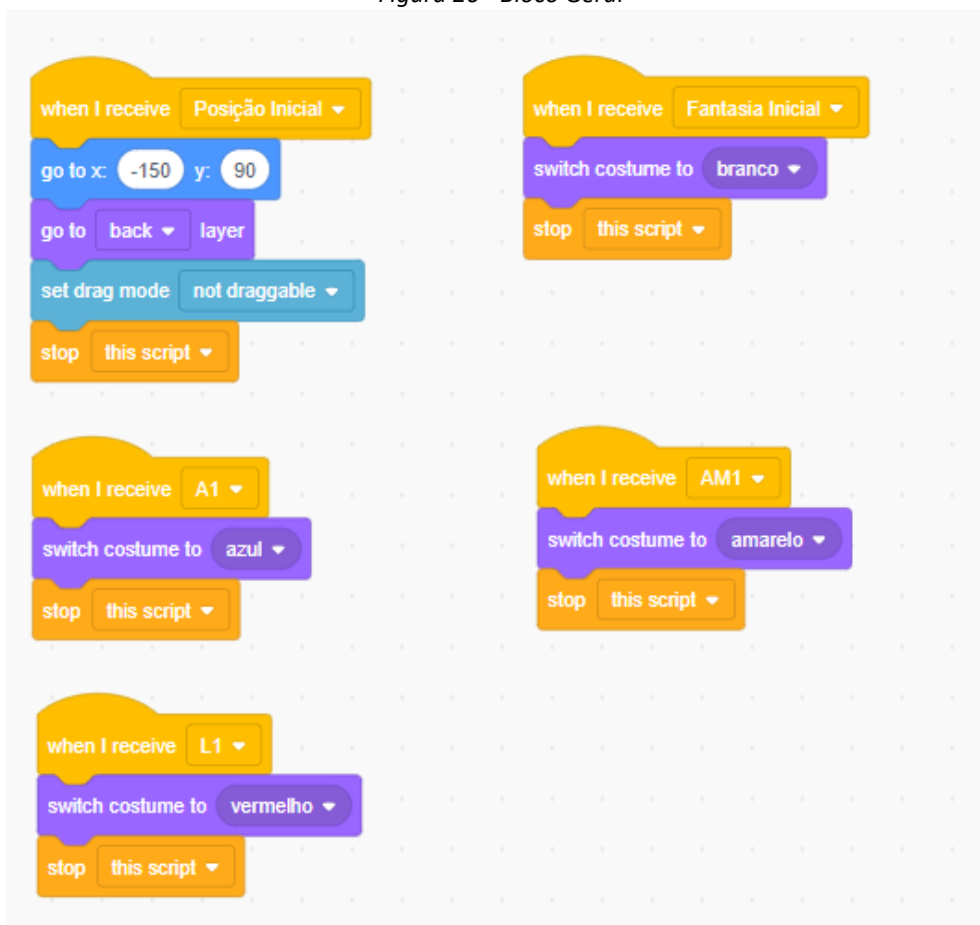


Fonte: Autora

A solução completa no *Scratch* pode ser acessada através do seguinte link:
<https://scratch.mit.edu/projects/610717909/>

Nesse passo é esperado que os alunos questionem as etapas de montagem do algoritmo disponibilizado, tanto quanto as cláusulas estruturais quanto ao raciocínio utilizado para a solução. E para cada questionamento, o professor deve realizar explicação relacionando cada linha de código com situações práticas para que a implementação se torne cada vez mais parte das soluções que forem desenvolver para os mais diversos problemas que compreendem situações do cotidiano.

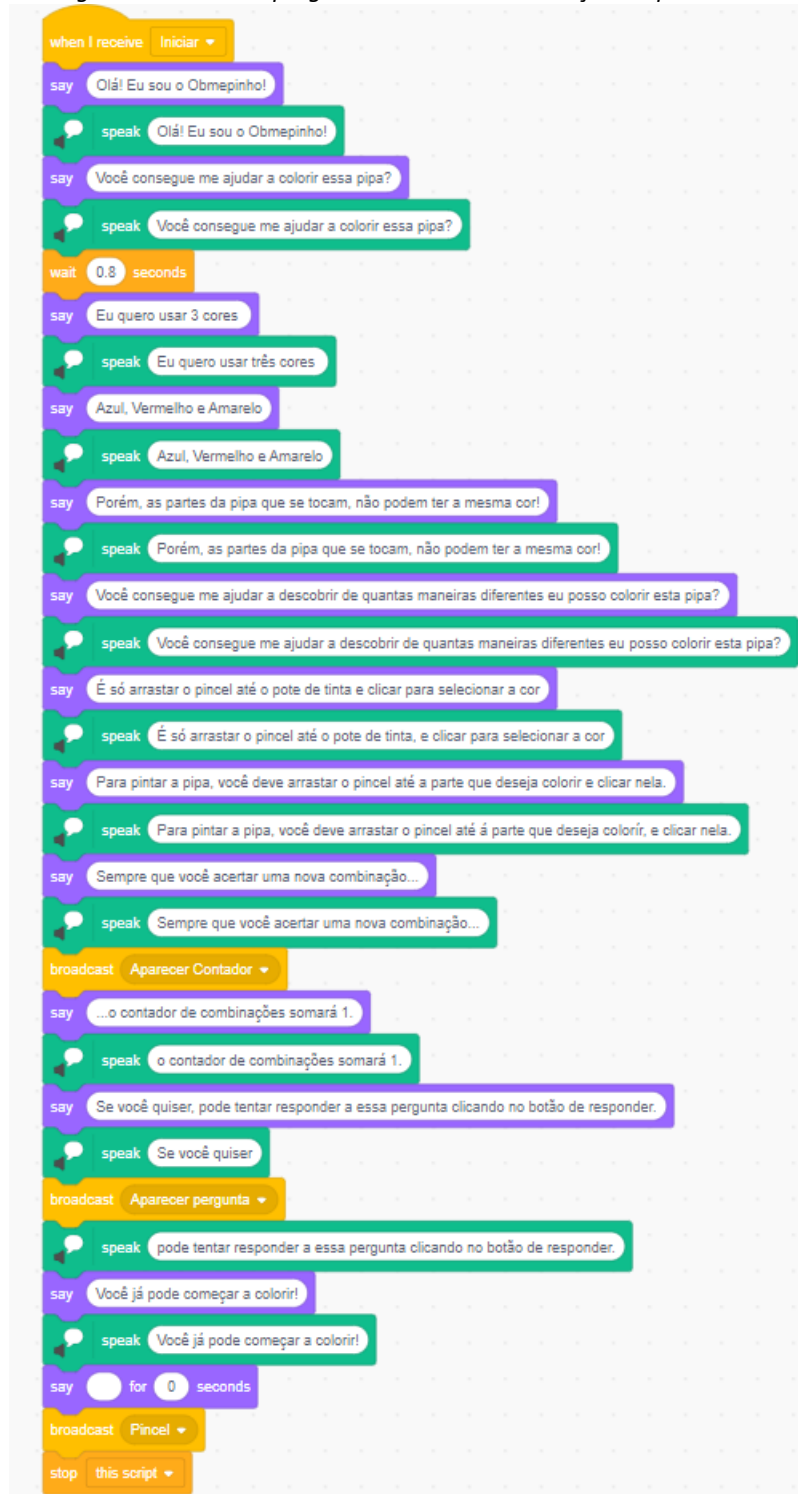
Figura 26 - Bloco Geral



Fonte: Autora

Na Figura 27, é possível visualizar as frases e perguntas para que se possa entender a situação problema e orientar os próximos passos para que a solução seja executada. Além disso, são incluídas ações para contador e pincel que são elementos participantes da solução. O contador irá armazenar a quantidade de tentativas corretas e o pincel servirá para indicar a área da pipa que deve ser pintada com uma das cores disponíveis.

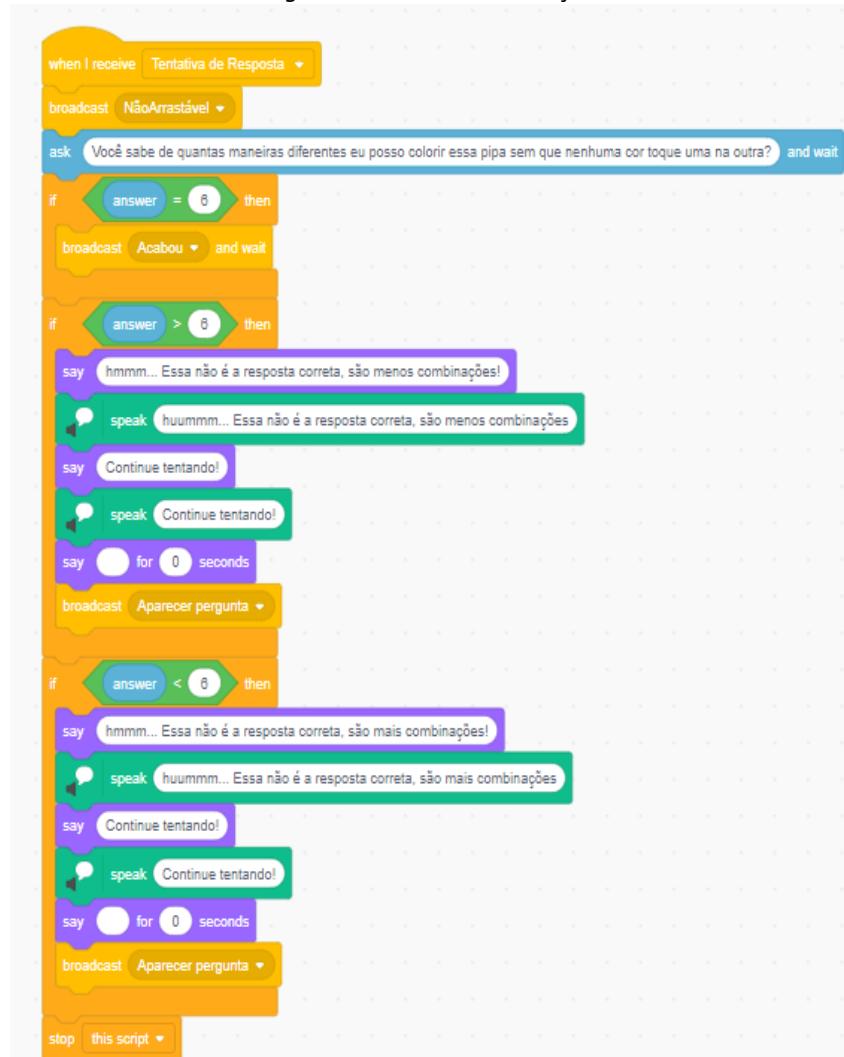
Figura 27 - Frases e perguntas iniciais de orientação ao problema



Fonte: Autora

Na Figura 28 é possível verificar o bloco que realiza a verificação das tentativas de respostas. Ou seja, a cada possibilidade de combinação de cores dos espaços em branco da pipa que esteja de acordo com as regras, um novo ponto deve ser inserido ao contador. Também é possível identificar as frases que servirão de resposta caso a combinação inserida não esteja de acordo com os requisitos estabelecidos.

Figura 28 - Frases de interação



Fonte: Autora

A seguir (Figura 29) é mostrado um trecho do código onde há uma interação entre o acerto de uma possibilidade e a continuidade das demais possibilidades. Além disso, é possível visualizar no final de cada bloco que ao acertar ou não uma possibilidade a figura da pipa é “reiniciada” para que novas possibilidades possam ser pintadas.

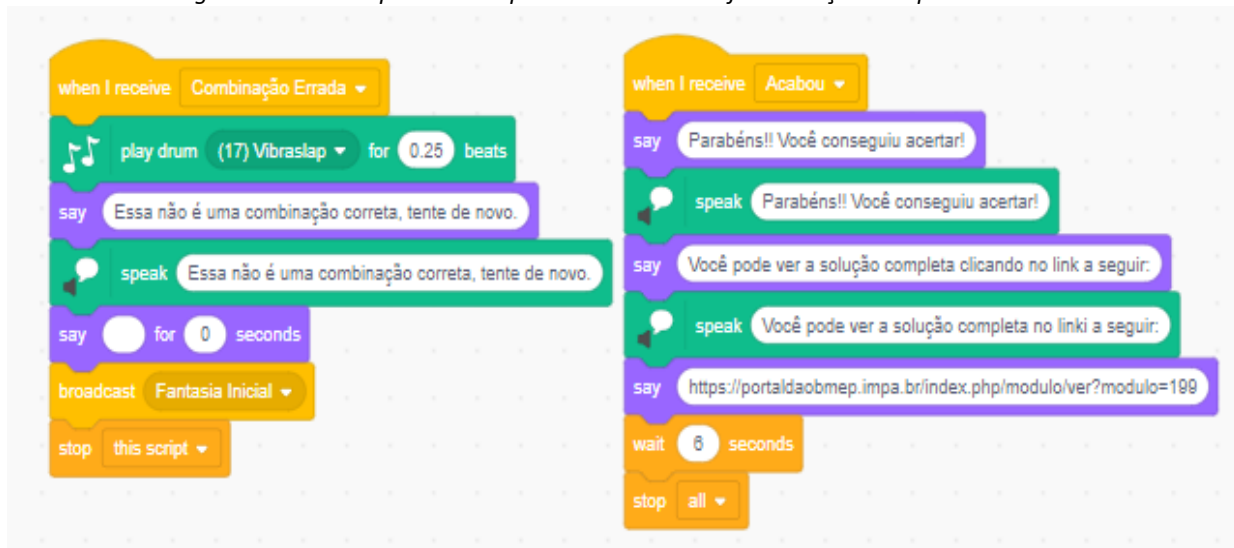
Figura 29 - Perguntas para verificação



Fonte: Autora

Na Figura 30 é possível verificar a configuração quanto aos efeitos de som e tempo pelo qual a animação ficará disponível em tela, tanto para frases que dizem respeito a possibilidades de acerto quanto para frases que indiquem possibilidades erradas.

Figura 30 - Frases que indicam possibilidade de verificar solução completa



Fonte: Autora

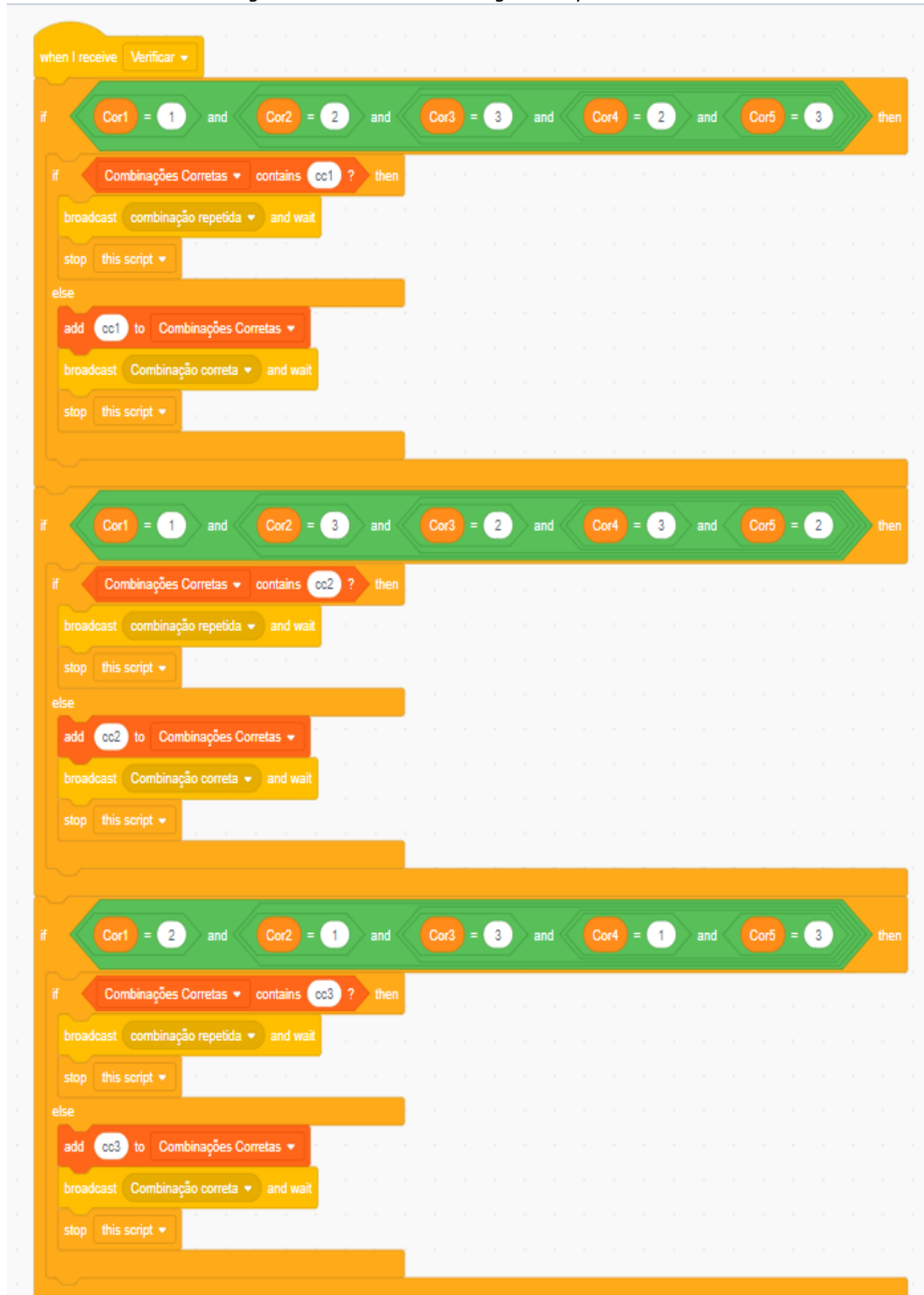
A seguir, nas Figura 31, 67 e 68 são demonstradas as verificações para as possibilidades de sequência de cores disponíveis para as opções fornecidas. Também é possível notar que as possibilidades são identificadas por: “Cor1”, “Cor2”, “Cor3”, “Cor4” e “Cor5”. E por isso, no início do bloco é chamada a verificação para uma dessas possibilidades, caso não seja nenhuma delas a execução chama o bloco “Combinação Errada”.

Figura 31 - Bloco com verificação das possibilidades



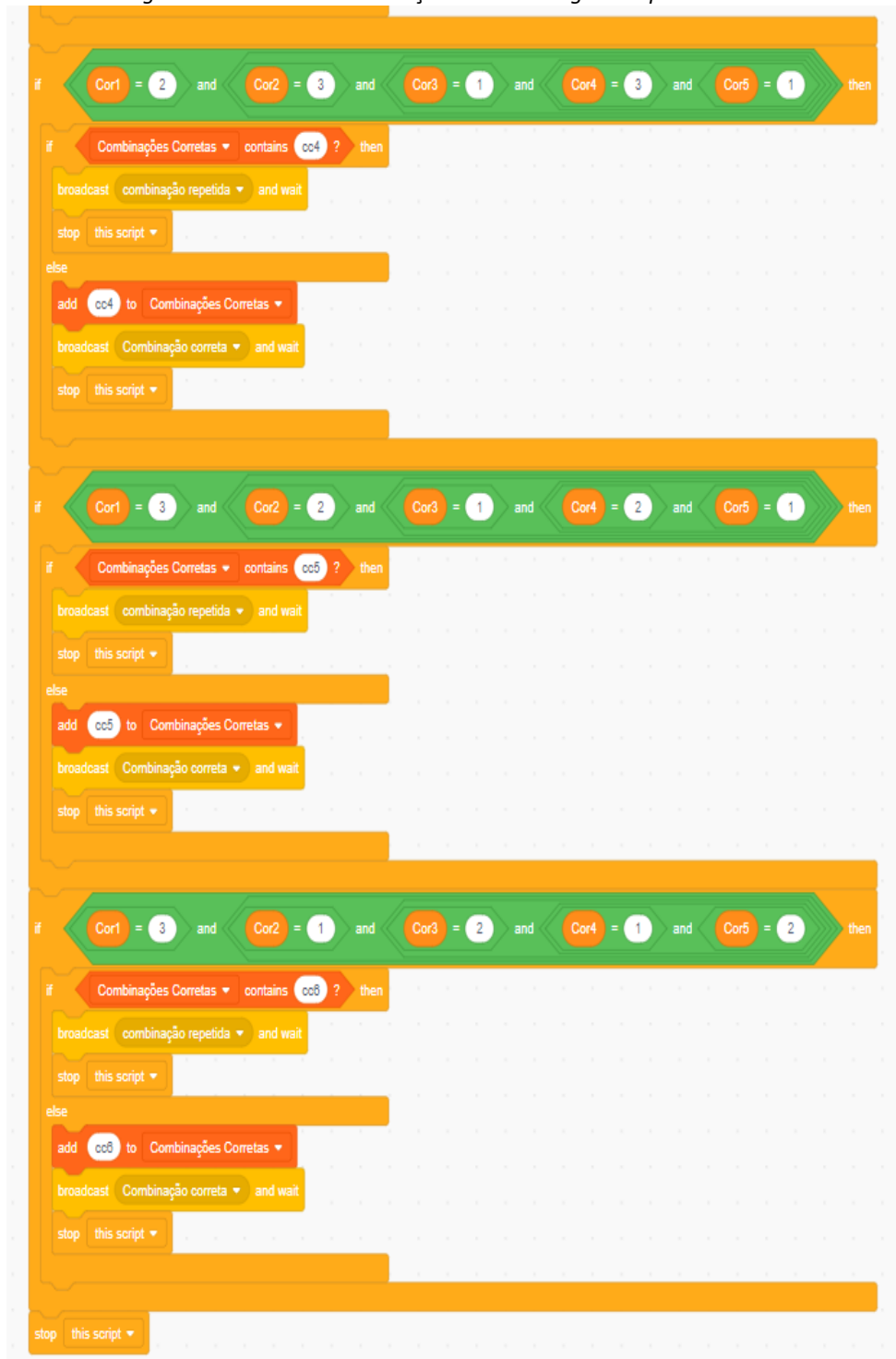
Fonte: Autora

Figura 32 - Bloco com montagem das possibilidades



Fonte: Autora

Figura 33 - Bloco de continuação com montagem de possibilidades

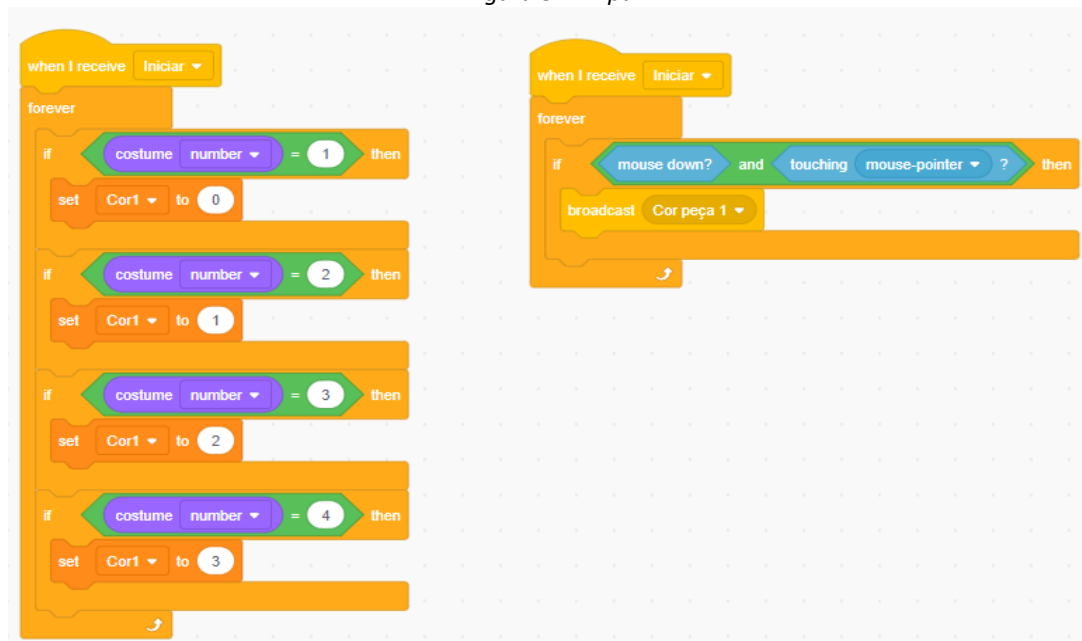


Fonte: Autora

No bloco a seguir (Figura 34) é demonstrada a localização dos espaços em branco da pipa. Nesse caso, quando o bloco for iniciado pela primeira ação, cada um dos espaços em branco passa a receber uma identificação para que depois sejam localizados e passem pela conferência de combinações. E no caso desse bloco, essa identificação está sendo realizada para a primeira ação, ou seja, a “Cor1”, por isso, no bloco existem as diferentes possibilidades de lugares que a “Cor1” pode estar localizada. Além disso, o bloco ao lado sinaliza

a localização do pincel, mostrando que o mesmo se encontra posicionado na “Cor1”.

Figura 34 - Pipa



Fonte: Autora

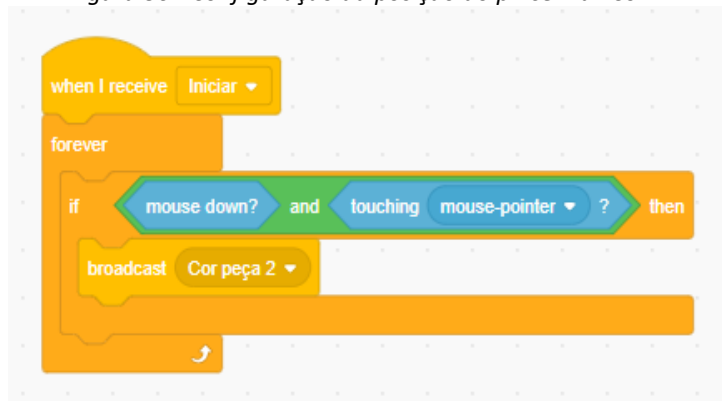
Após identificada a localização da “Cor1”, ou seja, a peça que receberá a “Cor1” e qual será essa cor (azul, vermelho ou amarelo), o próximo bloco (Figura 35), demonstra a identificação da próxima cor, no caso, a “Cor2”, que também deve ser identificada para posterior localização. Na Figura 36, é encontrado o bloco para configuração da posição do pincel, indicando que essa posição corresponde ao segundo espaço com a configuração da “Cor2”.

Figura 35 - Configuração da “Cor2”



Fonte: Autora

Figura 36 - Configuração da posição do pincel na “Cor2”



Fonte: Autora

Na Figura 37, é demonstrado o bloco que contém a configuração da posição da rabiola na pipa. Ou seja, dentro da limitação do quadro no qual acontecem as ações, a rabiola ocupa a posição determinada pelas variáveis “x” e “y”. Além disso, a configuração do bloco também indica que a rabiola é um elemento que não é arrastável.

Figura 37 - Rabiola



Fonte: Autora

O bloco a seguir (Figura 38) contém as instruções referentes ao “balde de tinta vermelha”, indicando sua posição dentro do quadro através das variáveis “x” e “y”. Além disso, após clicar sobre ele o pincel assume a cor referente ao conteúdo do balde, no caso, “vermelho”.

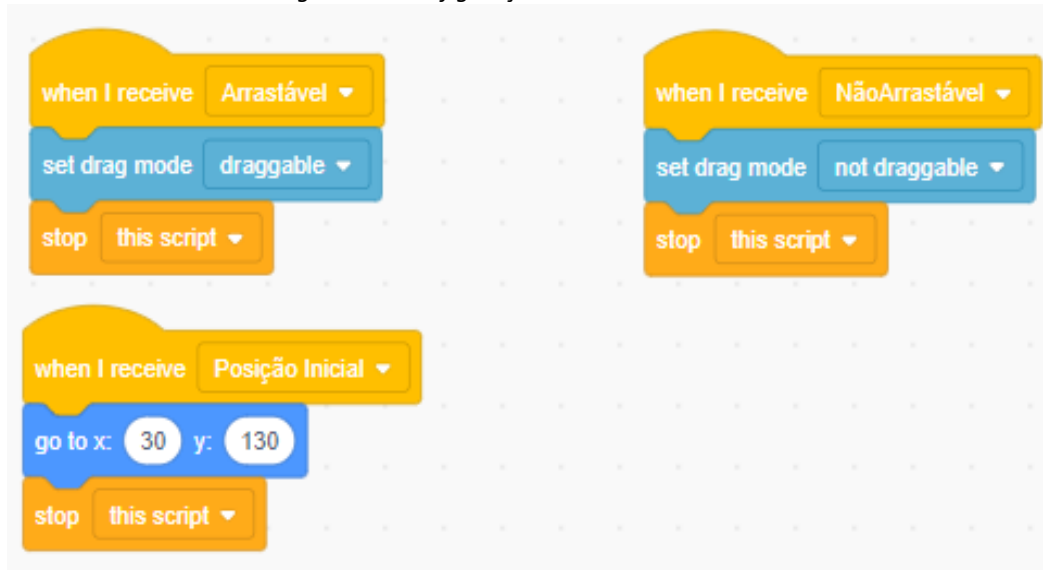
Figura 38 - Configuração da posição do pincel na “Cor2”



Fonte: Autora

O bloco a seguir (Figura 39) contém as instruções referentes ao “balde de tinta azul”, indicando sua posição dentro do quadro através das variáveis “x” e “y”. Além disso, após clicar sobre ele o pincel assume a cor referente ao conteúdo do balde, no caso, “azul”.

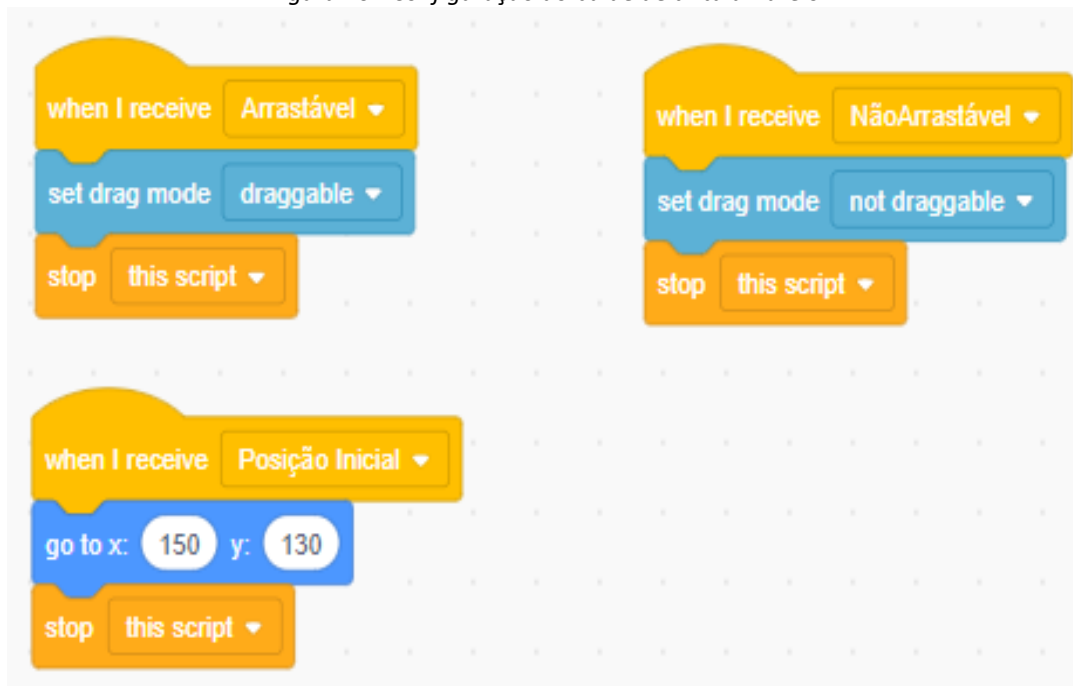
Figura 39 - Configuração do balde de tinta azul



Fonte: Autora

O bloco a seguir (Figura 40) contém as instruções referentes ao “balde de tinta azul”, indicando sua posição dentro do quadro através das variáveis “x” e “y”. Além disso, após clicar sobre ele o pincel assume a cor referente ao conteúdo do balde, no caso, “amarelo”.

Figura 40 - Configuração do balde de tinta amarelo

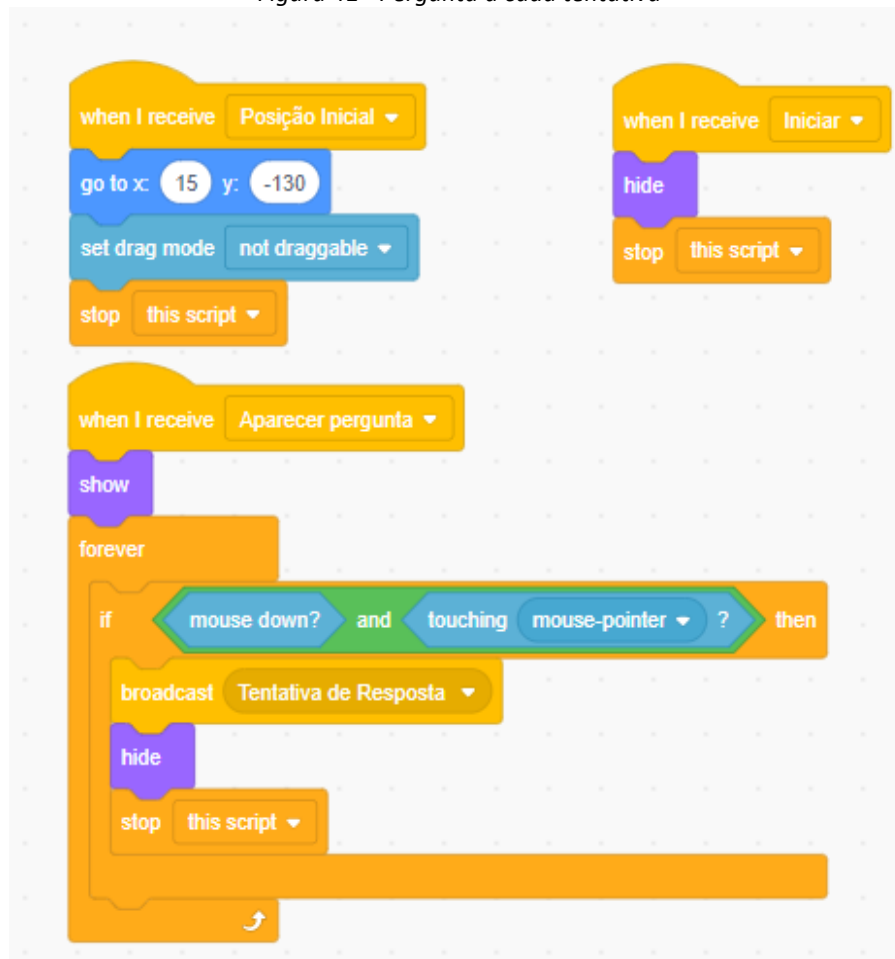


Fonte: Autora

No bloco a seguir (Figura 41), é demonstrada a configuração da pergunta que é feita para realização da tentativa de combinações. Nesse caso, é demonstrada a localização da pergunta no quadro através das variáveis “x” e “y” e que não deve ser arrastável. Além disso, é configurado que a pergunta deve aparecer na tela (através da ação “show”) e após iniciar as tentativas (identificada ação em

tela através da ação "touching") a pergunta deve desaparecer da tela (através da ação "hide").

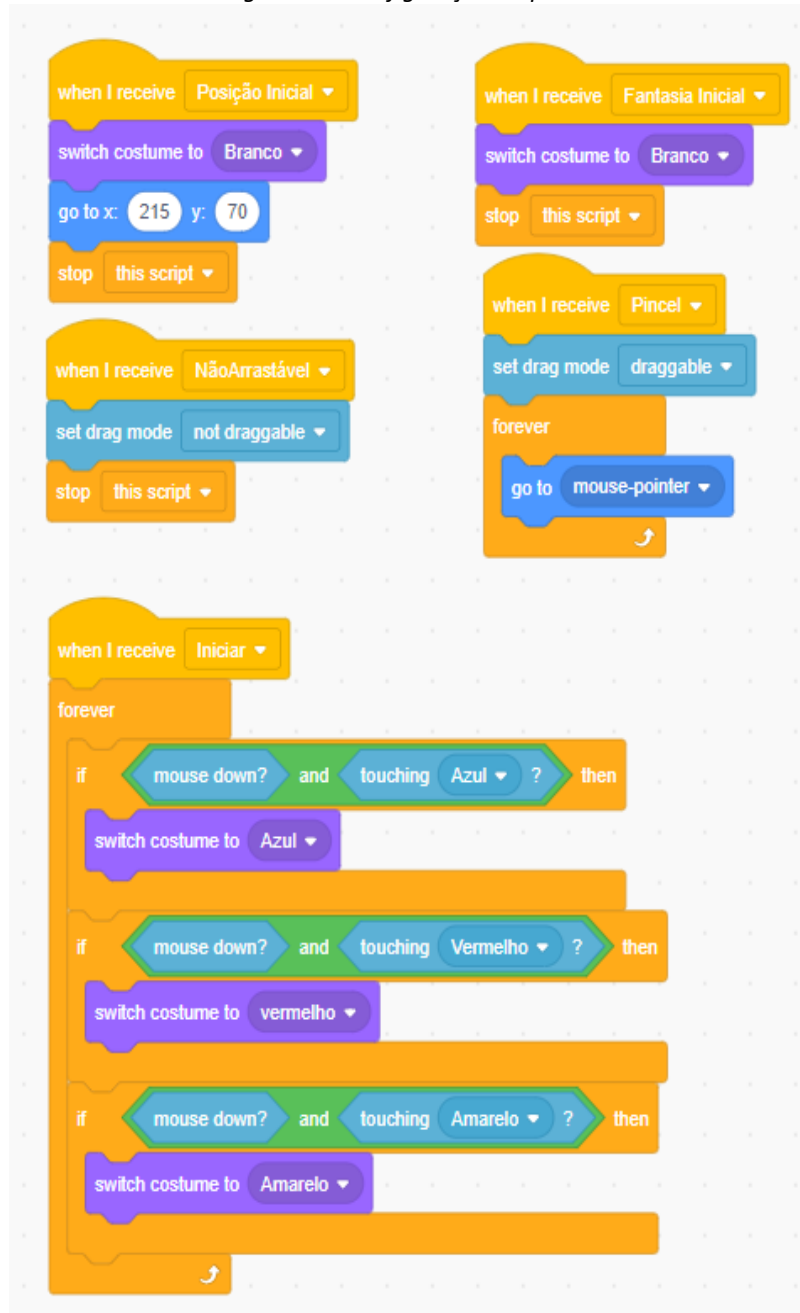
Figura 41 - Pergunta a cada tentativa



Fonte: Autora

De acordo com a Figura 42, pode ser verificada a configuração do pincel. É possível visualizar que a posição inicial do pincel é dada pelas variáveis "x" e "y" e que inicialmente encontra-se em branco, ou seja, sem tinta. Quando é permitido que as tentativas para pintura da pipa começam, o pincel torna-se arrastável para que possa assumir as cores "azul", "vermelho" ou "amarelo" para colorir os espaços em branco da pipa. A troca de cores é identificada através do comando "switch costume" após ser identificada pelo comando "touching", conforme demonstrado no bloco da figura abaixo.

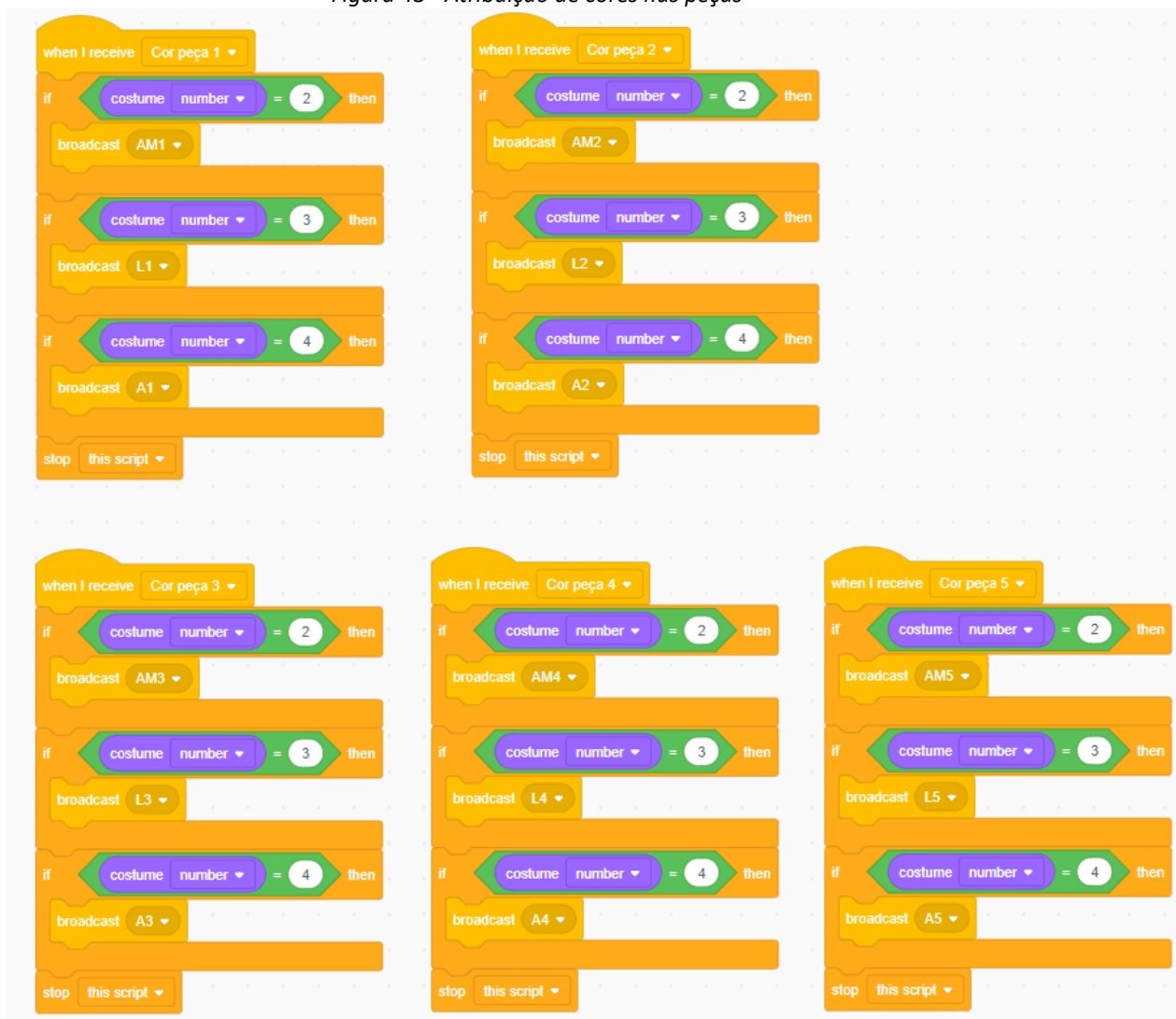
Figura 42 - Configuração do pincel



Fonte: Autora

A seguir, é demonstrado a atribuição de cores a cada uma das peças, sendo estas identificadas por “Cor peça 1”, “Cor peça 2”, “Cor peça 3”, “Cor peça 4” e “Cor peça 5”, em cada um dos 5 blocos detalhados na Figura 43. Além disso, o comando “costume” detalhe a cor que está sendo atribuída para a peça e onde essa peça está localizada, para que posteriormente, essa combinação de cores na localização que foram atribuídas seja verificada dentro das possibilidades permitidas.

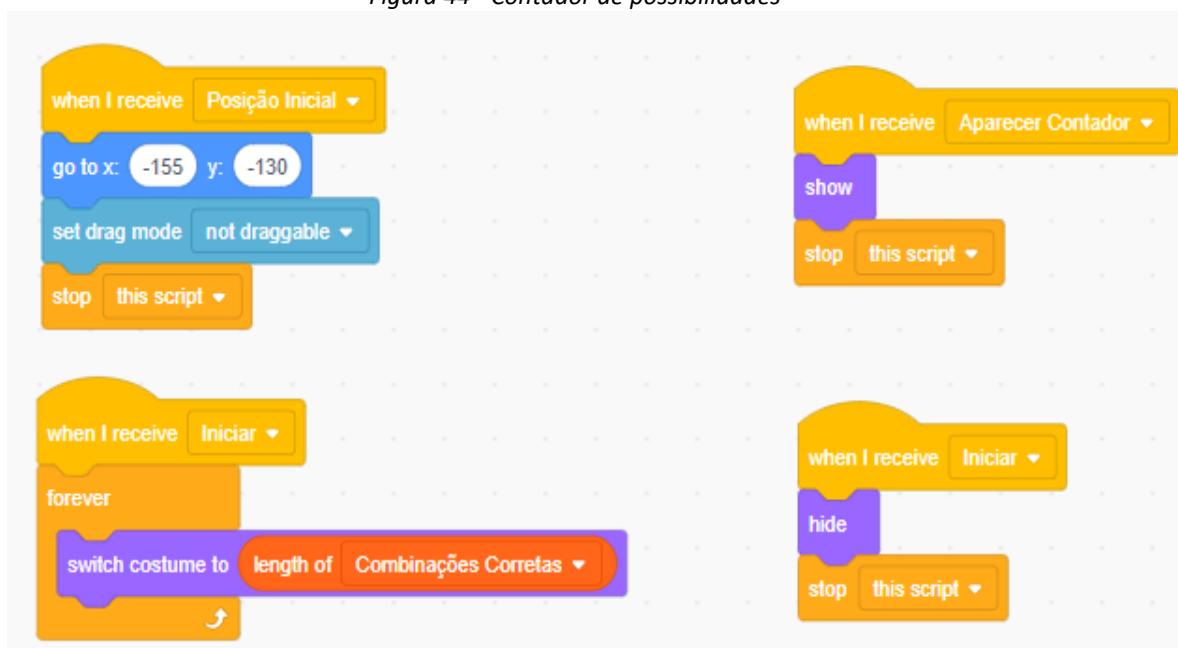
Figura 43 - Atribuição de cores nas peças



Fonte: Autora

A cada tentativa, é verificada se é uma possibilidade válida de acordo com a regra que partes com cores iguais não podem estar lado a lado na pintura da pipa. Se a tentativa for correta, é registrado um ponto no contador de possibilidades (quantidade de combinações). Na Figura 44, é possível visualizar as configurações para o Contador, que está localizado através das variáveis “x” e “y” e não é arrastável, ou seja, tem uma posição fixa. Além disso, ele aparece através do comando “show” quando a solução começa a ser executada, pois, anteriormente a isso o comando “hide” está sendo executado para que ele não apareça em tela. Também é possível observar por meio do comando “switch costume” que o placar vai sendo alterado a cada possibilidade válida através do gatilho do comando “length of” que contabiliza no contador “Combinações Corretas”.

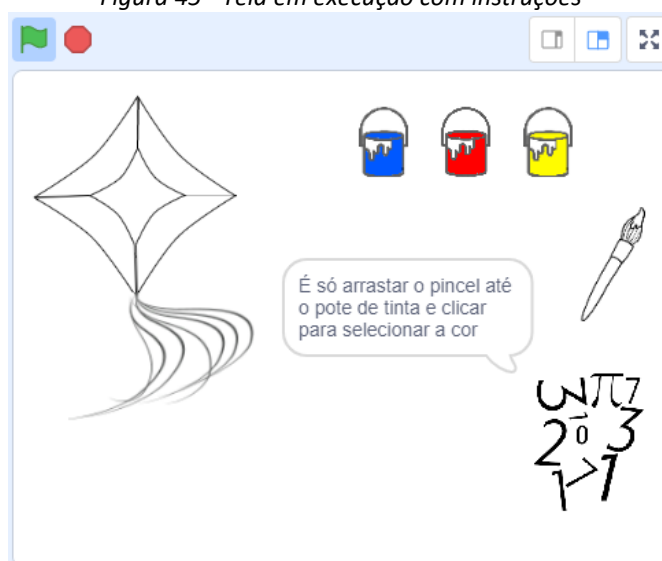
Figura 44 - Contador de possibilidades



Fonte: Autora

Na Figura 45, é demonstrado o programa (solução construída) no *Scratch* já em execução com o detalhamento das instruções para utilização do pincel a fim de colorir a pipa com uma possibilidade válida.

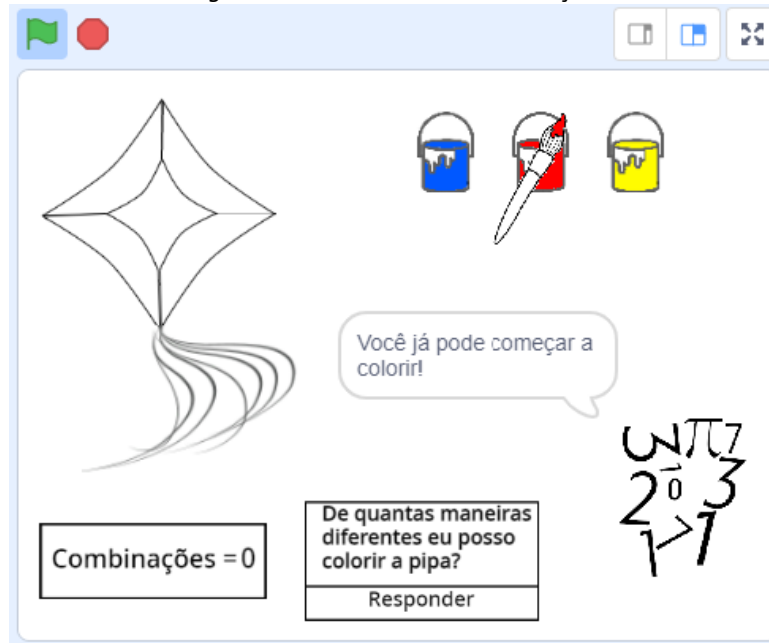
Figura 45 - Tela em execução com instruções



Fonte: Autora

A seguir, após as instruções passadas, o contador passa a aparecer em tela para que as possibilidades válidas passem a ser contabilizadas. Com isso, o pincel também é habilitado para que passe a assumir uma das cores e colorir os espaços em branco da pipa (Figura 46).

Figura 46 - Tela de início em execução



Fonte: Autora

Na Figura 47 é demonstrado a tela com uma das possibilidades corretas já executada e o contador tendo somado +1. Após isso, os espaços pintados da pipa são reiniciados (em branco) para que uma nova possibilidade seja pintada.

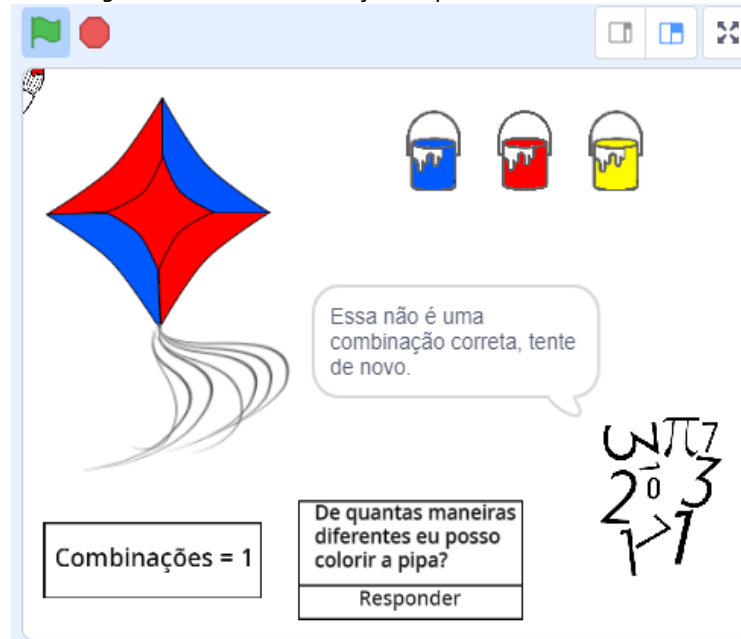
Figura 47 - Tela em execução de possibilidade correta



Fonte: Autora

A seguir é possível visualizar na Figura 48 um exemplo de uma possibilidade inválida, nesse caso, é exibida a mensagem indicando que se trata de uma possibilidade incorreta, não é somado ponto ao contador e a os espaços da pipa serão reiniciados para nova pintura.

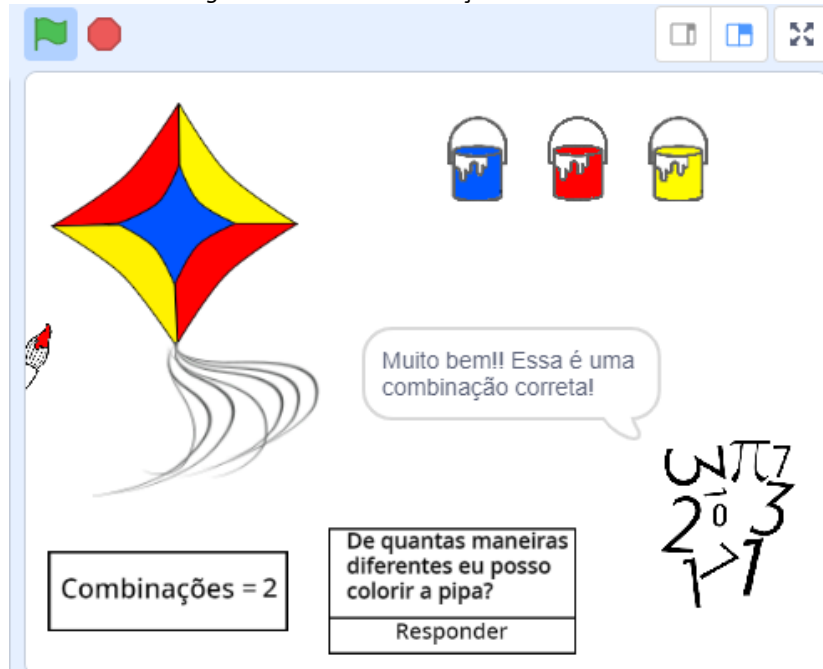
Figura 48 - Tela em execução de possibilidade incorreta



Fonte: Autora

Na Figura 49 é demonstrado a tela com mais uma das possibilidades corretas já executada e o contador tendo somado +1, nesse caso, totalizando “Combinações = 2”. Após isso, os espaços pintados da pipa são reiniciados (em branco) para que uma nova possibilidade seja pintada.

Figura 49 - Tela em execução com contador



Fonte: Autora

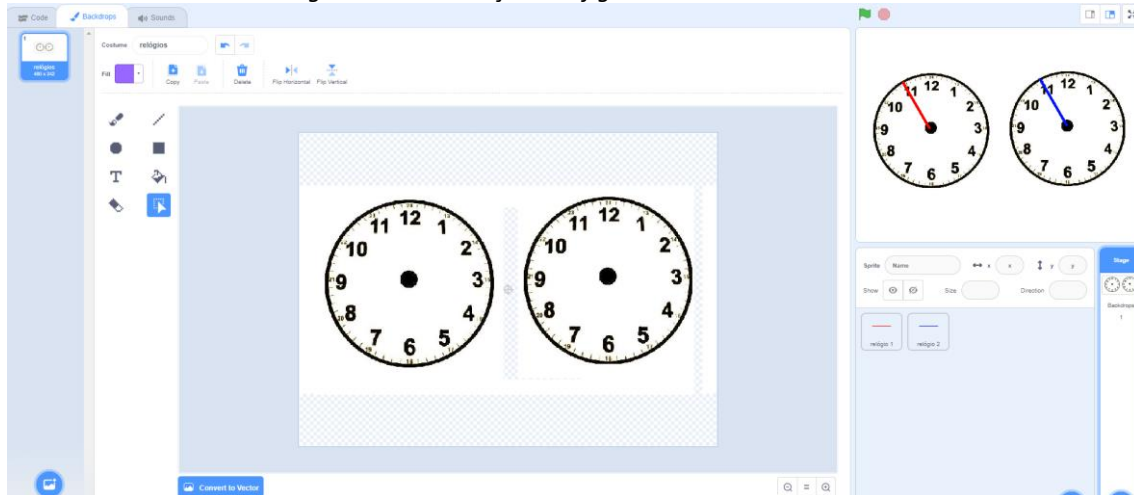
A solução completa no *Scratch* pode ser acessada através do seguinte link:
<https://scratch.mit.edu/projects/610716639/>

Nesse passo é esperado que os alunos questionem as etapas de montagem do algoritmo disponibilizado, tanto quanto as cláusulas estruturais quanto ao raciocínio utilizado para a solução. E para cada questionamento, o professor deve realizar explicação relacionando cada linha de código com situações práticas para que a implementação torne-se cada vez mais parte das soluções que forem desenvolver para os mais diversos problemas que compreendem situações do cotidiano.

APÊNDICE H – Solução Scratch (Questão 4 - RELÓGIOS)

Na Figura 50 é demonstrada a construção das figuras dos relógios a serem utilizados na solução, sendo possível visualizar a localização de cada um dos relógios na tela através dos eixos “x” e “y” e dos ponteiros “vermelho” e “azul” que diferenciam o relógio correto e o relógio em atraso.

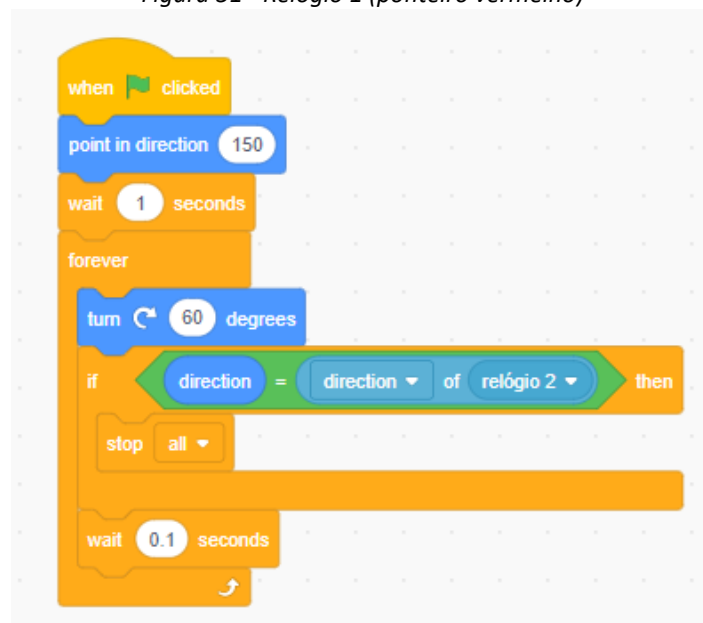
Figura 50 - Construção das figuras a serem utilizadas



Fonte: Autora

A seguir, na Figura 51 é demonstrado o código utilizado para as instruções referentes ao relógio 1 que contém o ponteiro vermelho e que não se encontra atrasado. É possível notar que o ponteiro está na posição inicial e que deve avançar 60° para que ao final da 1h tenham passado 60 minutos sem atraso. Além disso, é possível notar a cláusula onde o relógio deve parar, que é quando a direção do ponteiro estiver igual ao do relógio 2.

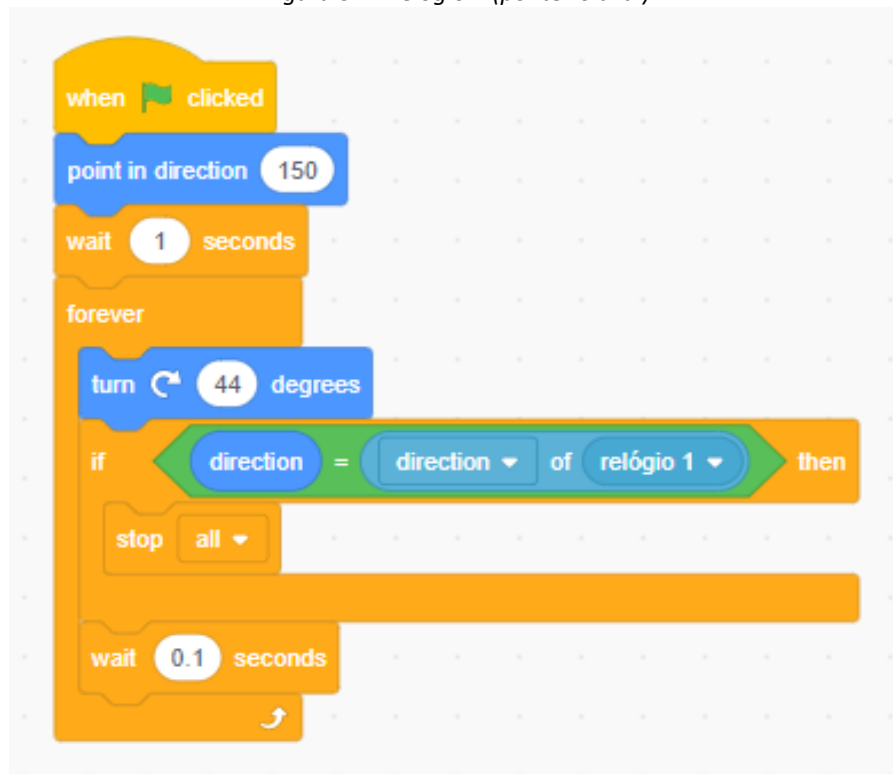
Figura 51 - Relógio 1 (ponteiro vermelho)



Fonte: Autora

Na Figura 52, é apresentado o código para o relógio 2 que contém o ponteiro azul e é identificado como o relógio em atraso. Por esse motivo, o relógio também está no mesmo ponto inicial, porém, ao invés de avançar 60° , avança 44° , pois, conforme mencionado no enunciado, consta com 16 minutos em atraso. Além disso, é possível notar a cláusula onde o relógio deve parar, que é quando a direção do ponteiro estiver igual ao do relógio 1. Dessa forma, os dois relógios irão parar no momento que chegarem ao mesmo horário, no caso 11h.

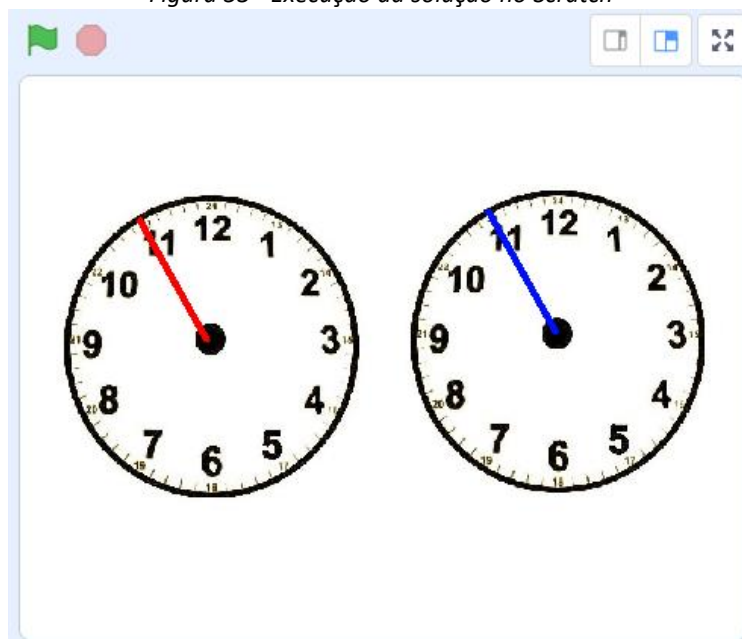
Figura 52 - Relógio 2 (ponteiro azul)



Fonte: Autora

A solução em execução no Scratch pode ser visualizada através da Figura 53, onde os dois relógios se encontram no mesmo ponto, no caso às 11h. Isso acontece, pois, no código foi implementado o raciocínio sobre a diferença entre os dois relógios, onde um está 16 minutos atrasados e o outro não. Por isso, um deve percorrer 44° e o outro 60° , e conforme mencionado na implementação, os dois relógios devem parar quando estiverem na mesma posição ao mesmo tempo.

Figura 53 - Execução da solução no Scratch



Fonte: Autora

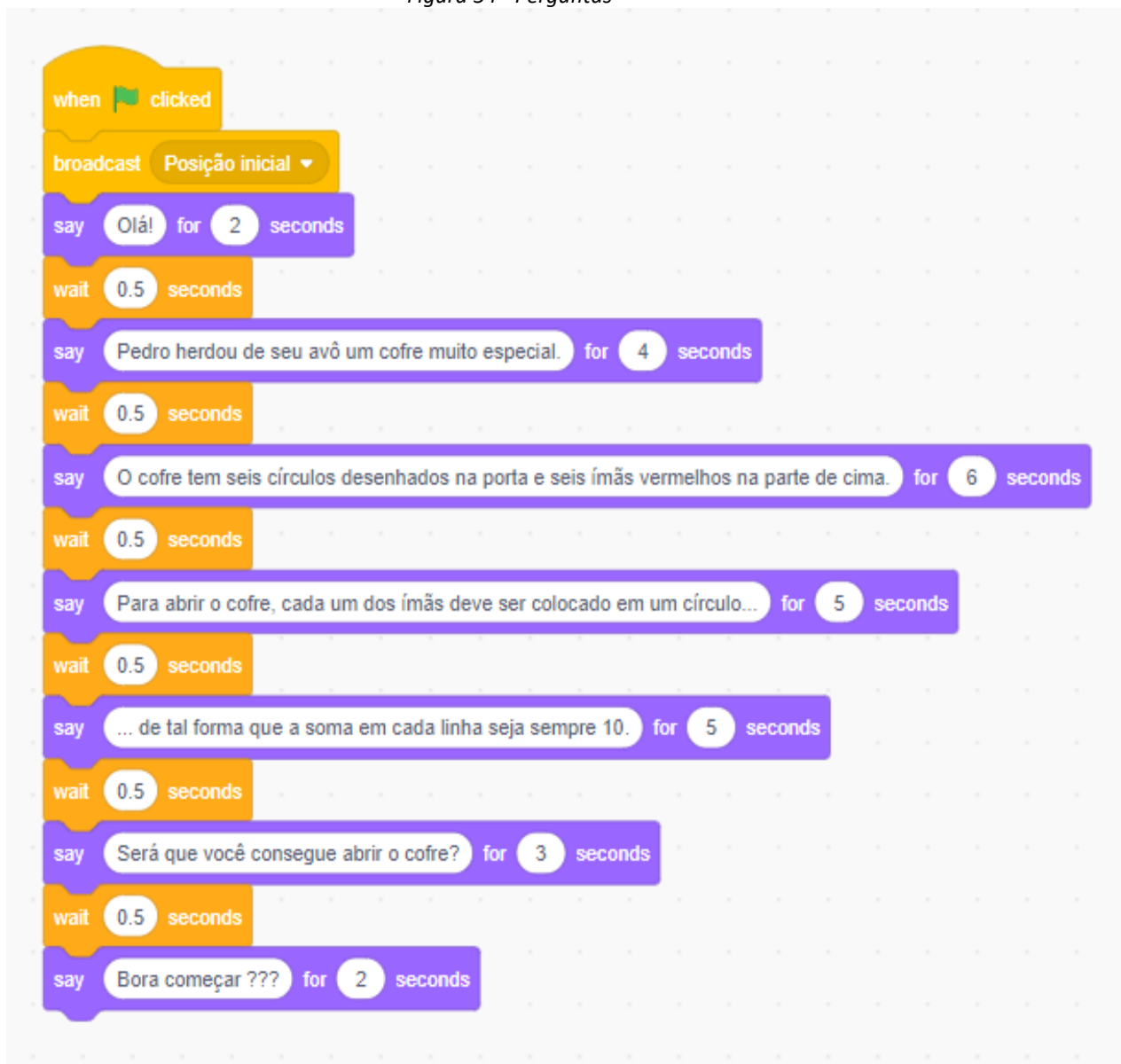
A solução completa no *Scratch* pode ser acessada através do seguinte link:
<https://scratch.mit.edu/projects/636989962/>

Nesse passo é esperado que os alunos questionem as etapas de montagem do algoritmo disponibilizado, tanto quanto as cláusulas estruturais quanto ao raciocínio utilizado para a solução. E para cada questionamento, o professor deve realizar explicação relacionando cada linha de código com situações práticas para que a implementação torne-se cada vez mais parte das soluções que forem desenvolver para os mais diversos problemas que compreendem situações do cotidiano.

APÊNDICE J – Solução Scratch (Questão 5 - COFRE)

Na Figura 54 é possível visualizar o bloco no qual são estruturadas as perguntas que conduzem a explicação do problema em torno do cofre de Pedro, ou seja, são uma sequência de frases que guiam o aluno a entender o problema mencionado.

Figura 54 - Perguntas



Fonte: Autora

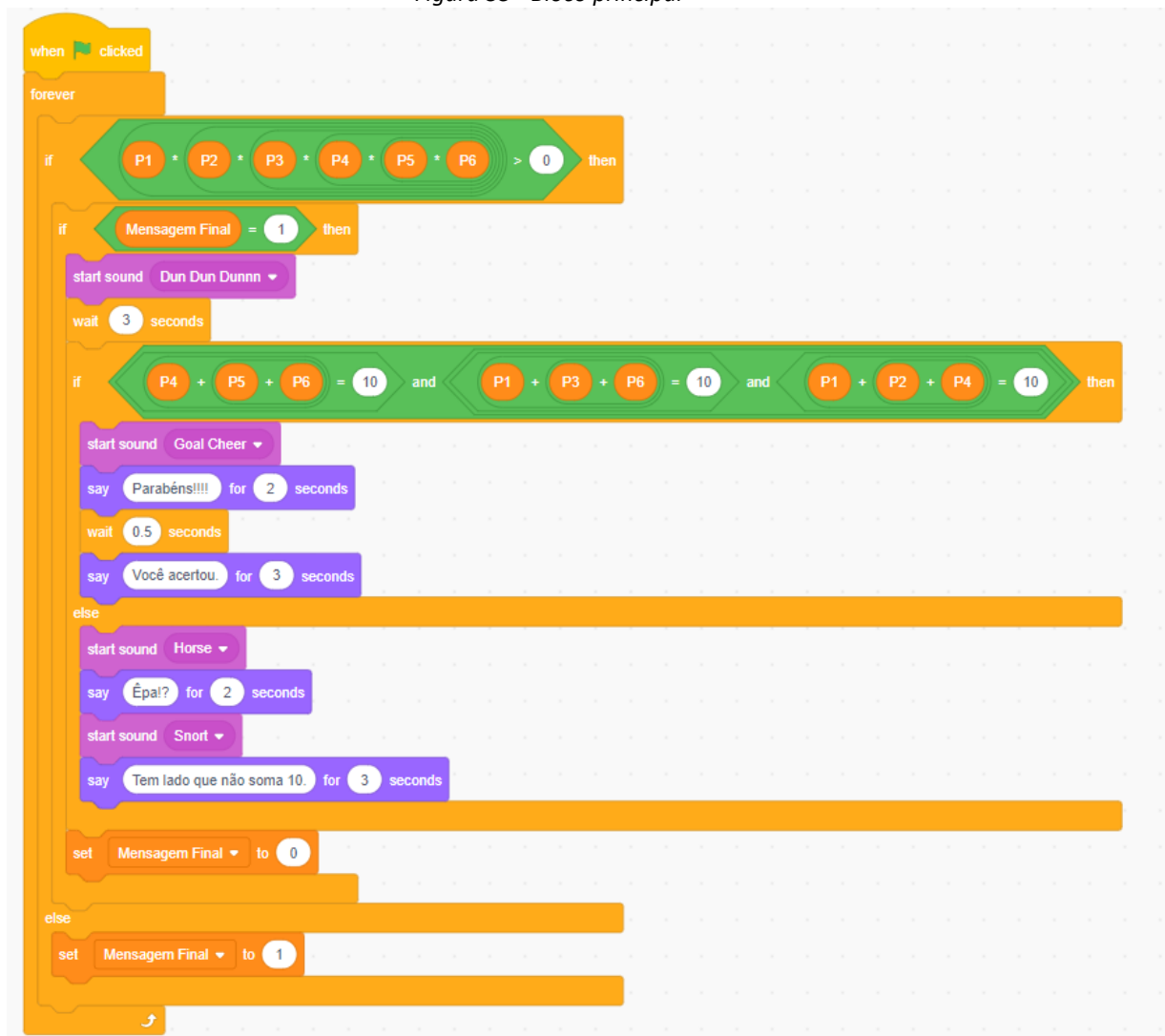
O bloco a seguir (Figura 55) representa as instruções para verificação dos espaços onde os imãs (botões) são posicionados. Portanto, é verificado se cada um dos círculos (espaços disponíveis) estão preenchidos, ou seja, se são maiores que zero. Após isso, é verificado se a soma de cada uma das linhas é igual a 10, caso seja, são exibidas as mensagens “Parabéns” e “Você acertou”. Caso não seja igual a 10, são exibidas as mensagens “Êpa!?” e “Tem lado que não soma 10”. Além disso, é importante notar que cada uma das linhas é identificada pelo conjunto de posições que contém, nesse caso:

Linha 1 – “P4” + “P5” + “P6”

Linha 2 - “P1” + “P3” + “P6”

Linha 3 - “P1” + “P2” + “P4”

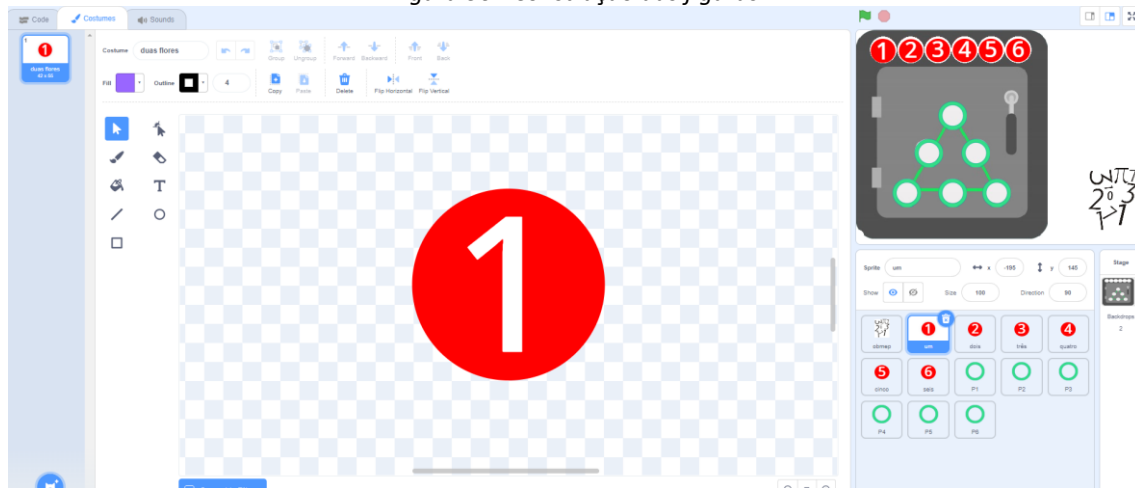
Figura 55 - Bloco principal



Fonte: Autora

A Figura 56 exibe a construção dos elementos visuais que compõem a ilustração da solução. Ou seja, nesse caso é possível visualizar o cofre, os botões e os ímãs.

Figura 56 - Construção das figuras



Fonte: Autora

No bloco seguinte (Figura 57) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

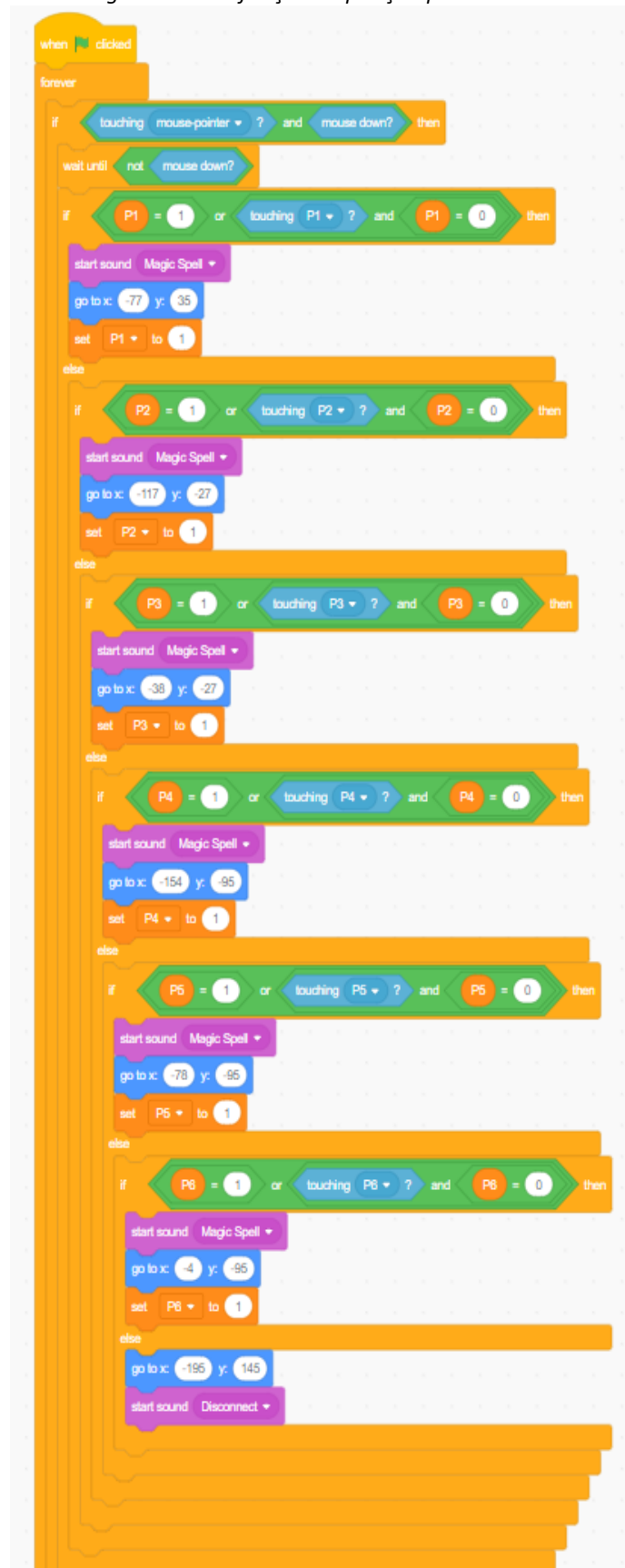
Figura 57 - Ímã “1”



Fonte: Autora

Na Figura 58 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 57) que pode ser visualizada no final desse bloco de instruções (Figura 58).

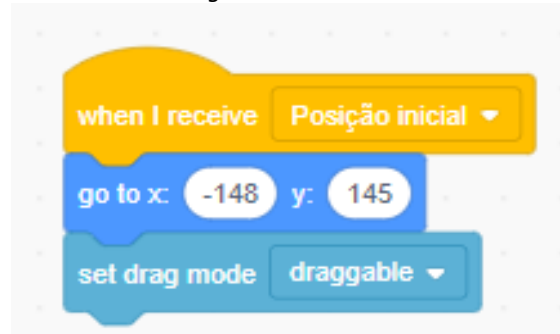
Figura 58 - Verificação da posição para imã "1"



Fonte: Autora

No bloco seguinte é (Figura 59) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

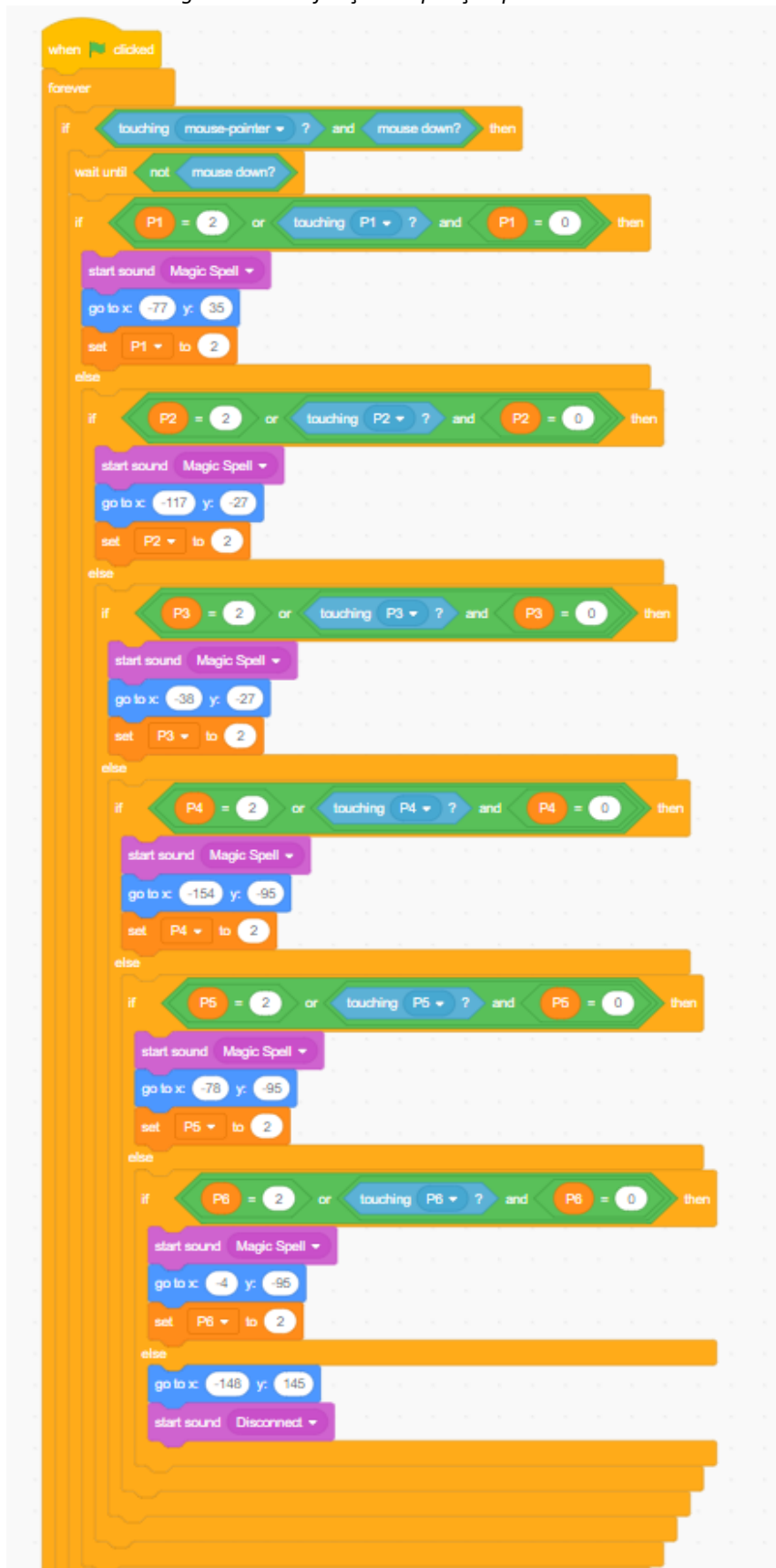
Figura 59 - Imã "2"



Fonte: Autora

Na Figura 60 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 59) que pode ser visualizada no final desse bloco de instruções (Figura 60).

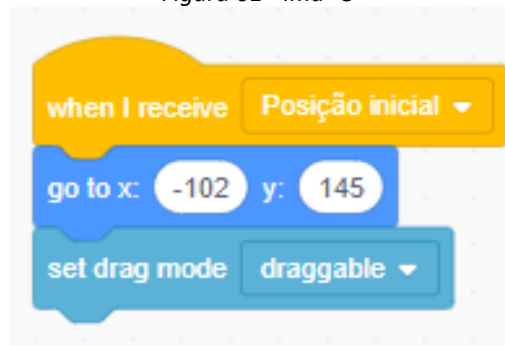
Figura 60 - Verificação da posição para imã "2"



Fonte: Autora

No bloco seguinte é (Figura 61) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

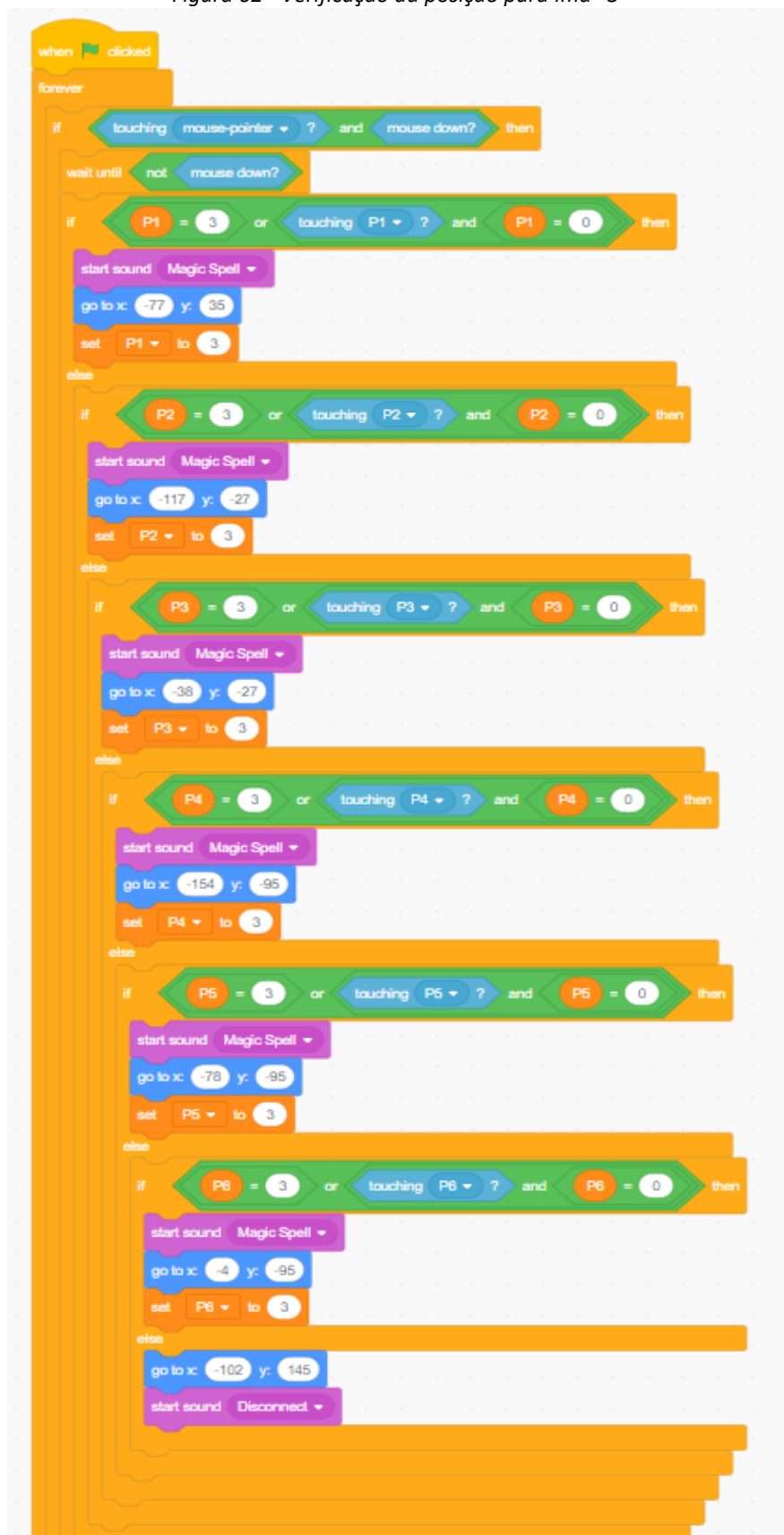
Figura 61 - Imã "3"



Fonte: Autora

Na Figura 62 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 61) que pode ser visualizada no final desse bloco de instruções (Figura 62).

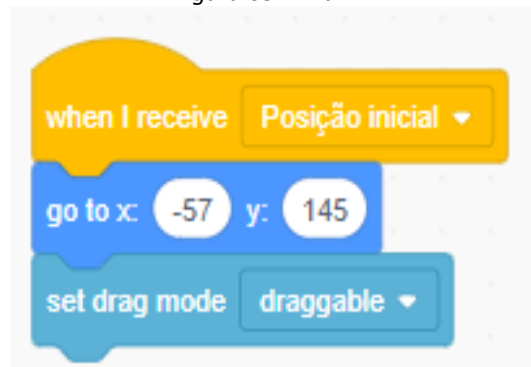
Figura 62 - Verificação da posição para imã "3"



Fonte: Autora

No bloco seguinte é (Figura 63) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

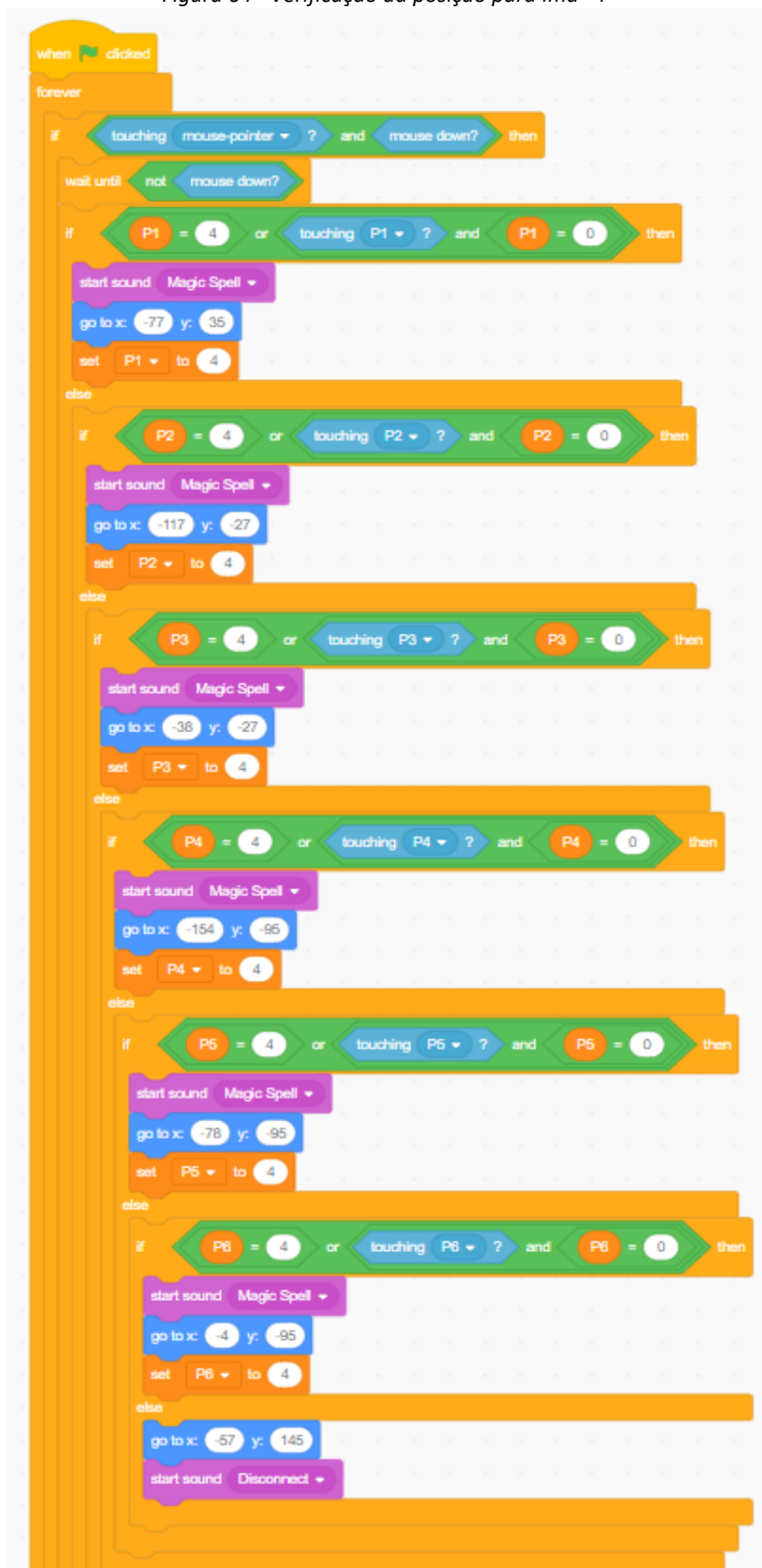
Figura 63 - Imã “4”



Fonte: Autora

Na Figura 64 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 63) que pode ser visualizada no final desse bloco de instruções (Figura 64).

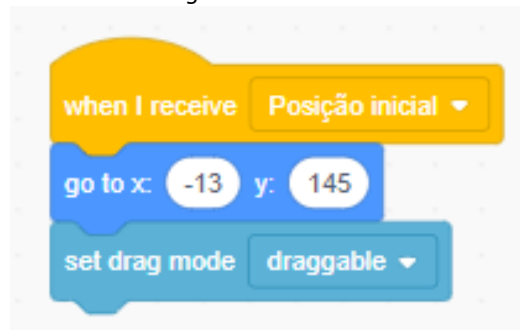
Figura 64 - Verificação da posição para imã "4"



Fonte: Autora

No bloco seguinte é (Figura 65) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

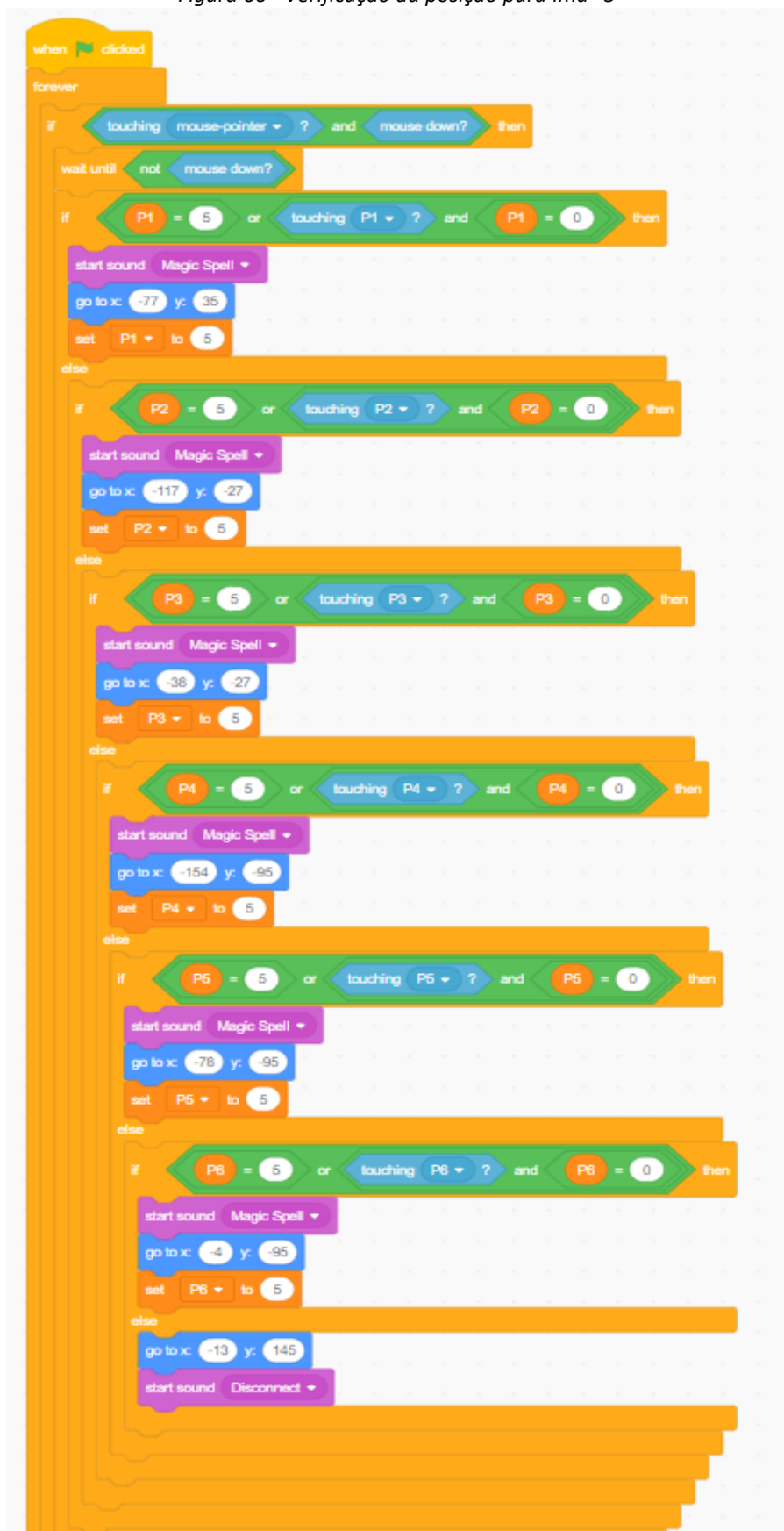
Figura 65 - Imã “5”



Fonte: Autora

Na Figura 66 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 65) que pode ser visualizada no final desse bloco de instruções (Figura 66).

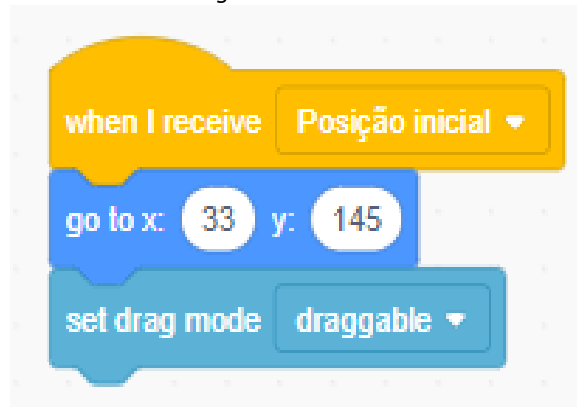
Figura 66 - Verificação da posição para imã "5"



Fonte: Autora

No bloco seguinte é (Figura 67) é possível visualizar a configuração da localização do “ímã 1” definido através das variáveis “x” e “y” e podendo ser arrastável para preencher os espaços em branco.

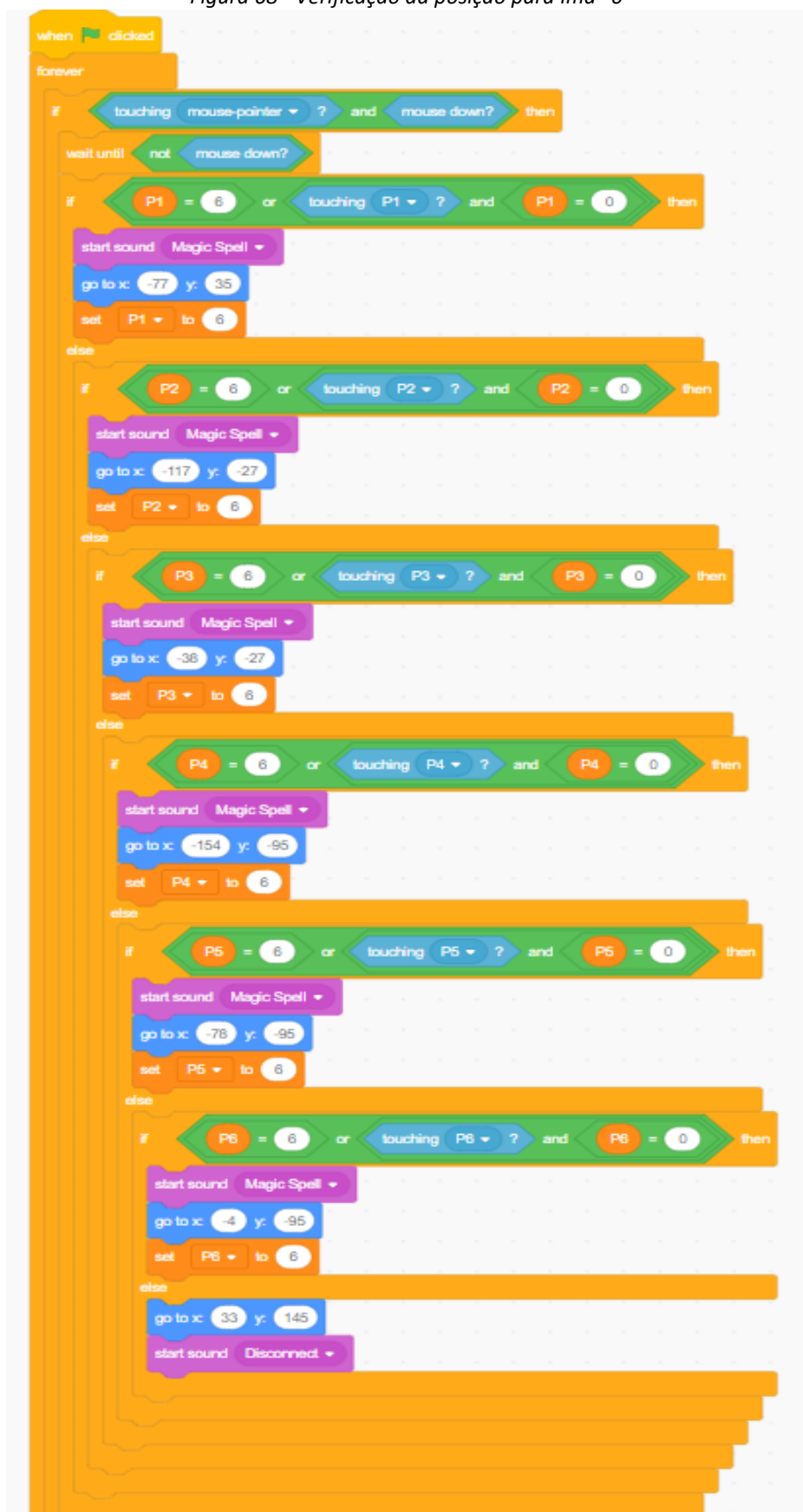
Figura 67 - Imã “6”



Fonte: Autora

Na Figura 68 são configuradas as instruções para verificação se cada um dos espaços estão preenchidos e com qual valor. Cada um dos espaços é identificado através das variáveis “x” e “y”. Caso o ímã seja colocado em um botão indevido o mesmo deve retornar para sua posição inicial (Figura 67) que pode ser visualizada no final desse bloco de instruções (Figura 68).

Figura 68 - Verificação da posição para imã "6"



Fonte: Autora

Na Figura 69 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P1”. Nesse caso, é verificado se foi colocado um dos imãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do imã colocado, se não permanece com valor “0”.

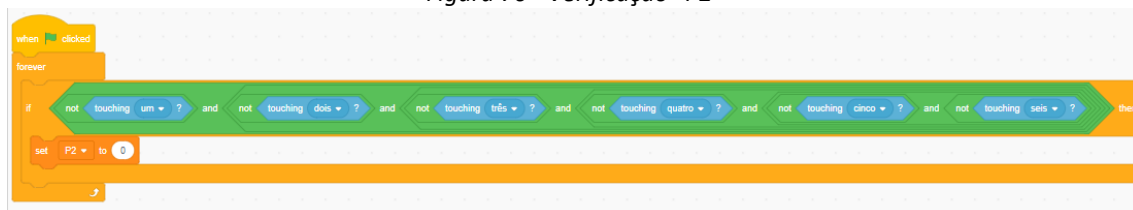
Figura 69 - Verificação "P1"



Fonte: Autora

Na Figura 70 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P2”. Nesse caso, é verificado se foi colocado um dos imãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do imã colocado, se não permanece com valor “0”.

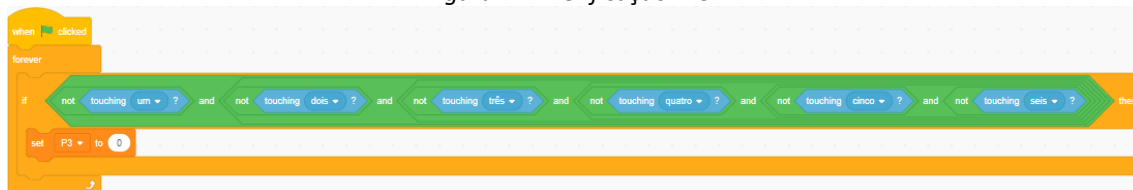
Figura 70 - Verificação "P2"



Fonte: Autora

Na Figura 71 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P3”. Nesse caso, é verificado se foi colocado um dos imãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do imã colocado, se não permanece com valor “0”.

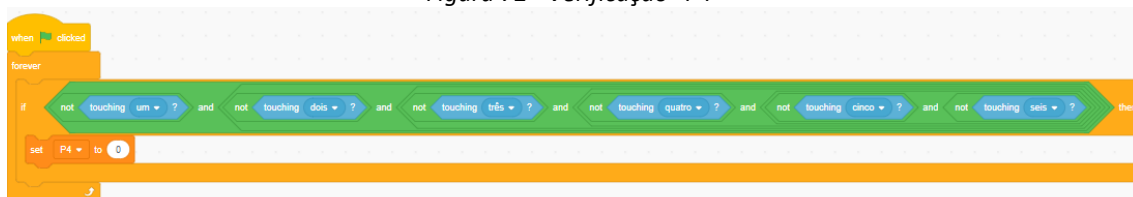
Figura 71 - Verificação "P3"



Fonte: Autora

Na Figura 72 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P4”. Nesse caso, é verificado se foi colocado um dos imãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do imã colocado, se não permanece com valor “0”.

Figura 72 - Verificação "P4"



Fonte: Autora

Na Figura 73 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P5”. Nesse caso, é verificado se foi colocado um dos ímãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do ímã colocado, se não permanece com valor “0”.

Figura 73 - Verificação "P5"



Fonte: Autora

Na Figura 74 é demonstrado o conjunto de instruções que seta o valor para o espaço em branco “P6”. Nesse caso, é verificado se foi colocado um dos ímãs disponíveis no espaço, se sim, o espaço passa a assumir o valor do ímã colocado, se não permanece com valor “0”.

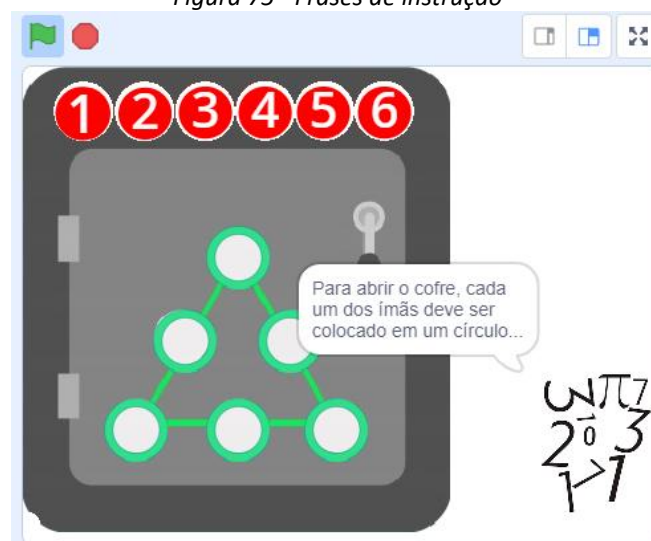
Figura 74 - Verificação "P6"



Fonte: Autora

De acordo com a Figura 75, é possível verificar a execução da solução construída no Scratch. Nesse caso, conforme visto no detalhamento das primeiras instruções está sendo executada a instruções que exibe as frases que guiam o aluno a entender o problema.

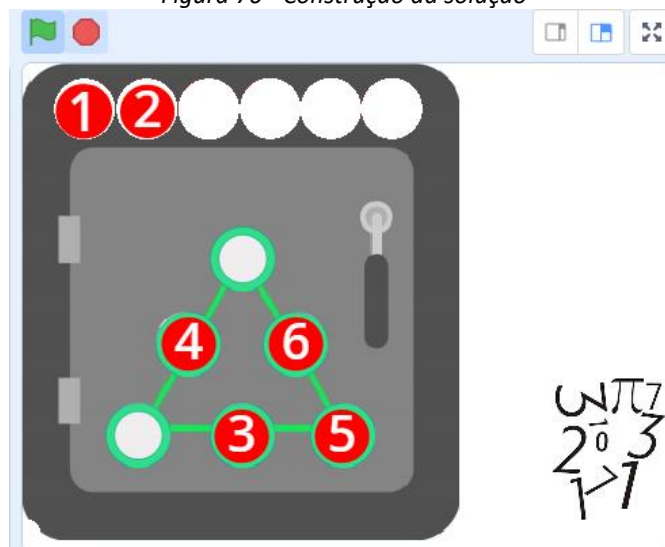
Figura 75 - Frases de instrução



Fonte: Autora

A seguir (Figura 76) é possível visualizar a execução da construção de uma tentativa de solução, onde os ímãs estão sendo colocados nos espaços em branco.

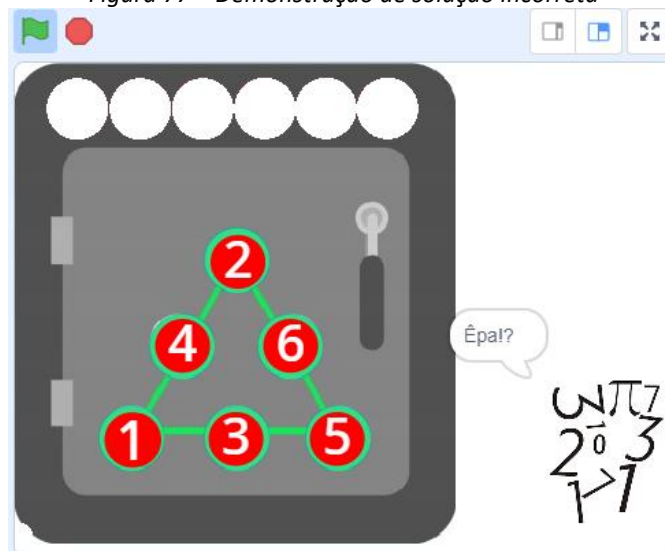
Figura 76 - Construção da solução



Fonte: Autora

Na Figura 77 é demonstrada a montagem de uma solução incorreta, onde conforme explicado trata-se da tentativa nas quais as linhas ou uma delas não tem o resultado 10 ao final da soma executada.

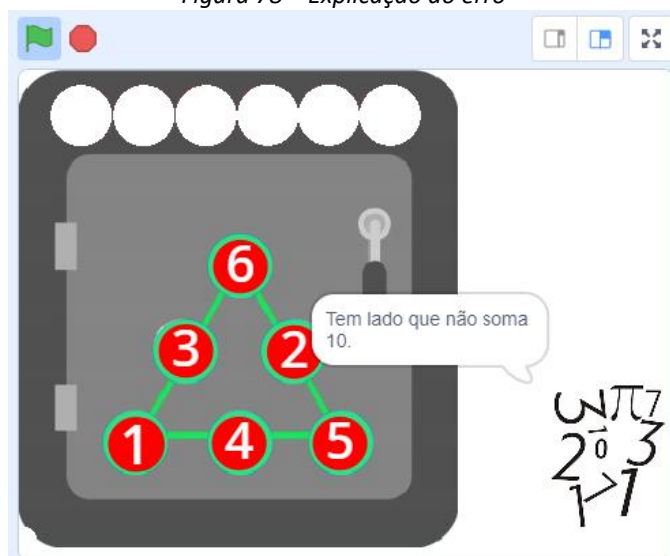
Figura 77 – Demonstração de solução incorreta



Fonte: Autora

Na Figura 78, está sendo executada a continuidade da explicação sobre a tentativa incorreta, onde é exibida a frase “Tem lado que não soma 10”, conforme explicado no início da solução e destacado ao realizar a explicação dos blocos de verificação quanto a essa regra.

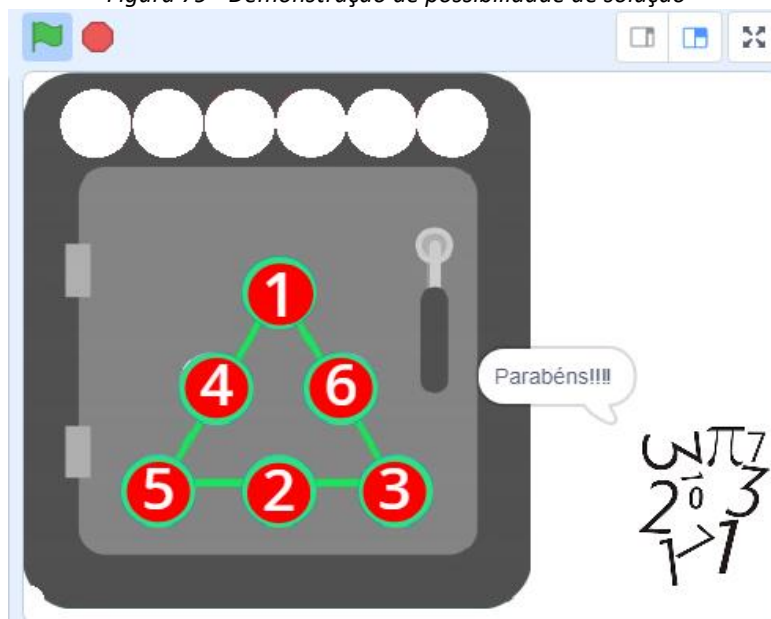
Figura 78 – Explicação do erro



Fonte: Autora

Para concluir, através da Figura 79 é possível visualizar uma tentativa correta em execução, ou seja, onde todas as linhas somam 10. Quando isso acontece é exibida a mensagem “Parabéns!!!!” (conforme instruções já detalhadas nos blocos de código), pois, o problema foi solucionado. É importante destacar que, existe mais de uma possibilidade de solução para esse problema, desde que o resultado da soma de cada uma das linhas seja 10 (conforme verificação demonstrada pela solução construída através dos blocos de códigos analisados).

Figura 79 - Demonstração de possibilidade de solução



Fonte: Autora

A solução completa no *Scratch* pode ser acessada através do seguinte link:
<https://scratch.mit.edu/projects/610713222/>

Nesse passo é esperado que os alunos questionem as etapas de montagem do algoritmo disponibilizado, tanto quanto as cláusulas estruturais quanto ao raciocínio utilizado para a solução. E para cada questionamento, o professor deve realizar explicação relacionando cada linha de código com situações práticas para que a implementação se torne cada vez mais parte das soluções que forem desenvolver para os mais diversos problemas que compreendem situações do cotidiano.