

Apostila

Programando com Alice





Sumário

1. Introdução	3
2. Histórico	4
3. Instalando o software Alice	5
4. Introdução ao ambiente Alice 2.0.....	8
4.1. Objetos	10
4.2. Métodos.....	11
4.3. Área de Edição e Funções Lógicas	11
5. Programando com Alice	12
5.1. Do in order	15
5.2. Do together	17
5.3. If/Else	18
5.4. Loop	20
5.5. While	21
5.6. For all in order	23
5.7. For all together	24
6. Eventos	27
6.1. Eventos Não-Interativos.....	28
6.2. Eventos Interativos	28
Trabalho Final	30
Apêndice A	31
Propriedades	31
Apêndice B	34
Importando figuras.....	34
Importando outros objetos.....	35
Apêndice C	36
Criando vídeos	36
Referências Bibliográficas	38



1. Introdução

Alice é um ambiente de programação tridimensional de fácil utilização no qual podem ser criadas animações e interações entre personagens e objetos lembrando muito jogos de vídeo game. Desenvolvido pela Universidade Carnegie Mellon, localizada na cidade de Pittsburgh, Pensilvânia, Estados Unidos, o objetivo do software Alice é ensinar os principais conceitos de programação aos alunos do Ensino Médio. Posteriormente, devido a sua facilidade e compreensão, o alvo de estudo ao Alice tornou-se a Lógica Computacional.

O programa Alice basicamente compreende um ambiente virtual denominado World (Mundo), onde as interações acontecem. A partir de elementos pré-programados que acompanham o ambiente, é possível criar histórias seguindo uma construção lógica. Cada objeto possui características próprias com classes básicas que podem ser modificadas e organizadas a maneira do programador. Trata-se de uma aplicação de uma linguagem orientada a objetos aos moldes das linguagens mais conhecidas como Java e C++, por exemplo.



2. Histórico

A primeira versão do software Alice foi disponibilizada para download em 1999 e já passou por inúmeras atualizações, porém, ainda é uma versão parcial. Inicialmente, o projeto era pequeno com o intuito de ensinar os conceitos básicos de programação aos alunos do Ensino Médio. Com o tempo, o projeto cresceu, conseguiu o financiamento de grandes empresas na área de software e, atualmente, conta com vários colaboradores, professores e pesquisadores não só da Universidade Carnegie Mellon, mas também provenientes de outras universidades. Os primeiros e principais financiadores do projeto são a Microsoft e a SunMicrosystems.

O software Alice é todo feito em Java. Atualmente, a versão 2.0 passou por novas atualizações e em junho de 2009 foi lançada a versão 2.2. Ainda aos moldes da versão anterior, foram corrigidos alguns bugs e, agora, conta com a possibilidade de transformar as animações criadas no Alice em vídeos com a extensão “.mov”. No período que compreende o final do ano de 2007 e início de 2008, a Universidade Carnegie Mellon recebeu um novo colaborador, a Electronic Arts (normalmente abreviada como EA). A Electronic Arts é a segunda maior empresa independente que atua na criação de jogos digitais. Com essa parceria, a inovação acrescida pela EA foi a implementação dos personagens do ambiente Alice. Assim, foi disponibilizada a utilização de personagens do jogo The Sims no projeto Alice. Em junho de 2009, uma versão Alice 3.0 Beta foi disponibilizada no site oficial do projeto Alice com os recursos acrescidos pela Electronic Arts, mas ainda são necessários muitos ajustes. Enquanto a versão final não sai, é recomendada a utilização das versões Alice 2.0 ou 2.2.

3. Instalando o software Alice

O software Alice é todo feito em Java e não há muitas restrições quanto a sua instalação. Inicialmente, é necessário que a máquina virtual Java esteja instalada para que o software Alice seja executado. Para isso, basta acessar o site da Sun Microsystems, www.sunmicrosystems.com, e realizar o download da máquina virtual Java mais recente ou daquela que for compatível com os requisitos do seu computador. O software Alice está disponível no site www.alice.org. Ao abrir o site, no lado esquerdo, abaixo, no menu downloads, escolha a opção Alice 2.0. As telas apresentadas a seguir possuem o intuito de auxiliá-lo(a) na instalação do software.



Figura 1: Página do software Alice.



Alice An Educational Software that teaches students computer programming in a 3D environment

● About Alice ● Downloads ● Teaching ● Community ● Publications ● Support

Download Alice v2.0

Windows

System requirements

- Windows Vista, XP, or 2000
- Intel Pentium II or equivalent processor
- A VGA graphics card capable of high (16 bit) color and 1024x768 resolution (3D video card recommended)
- 512MB of RAM (1GB recommended)
- A sound card

[Download Alice for PC](#)

Mac

OS X 10.4+

OS X 10.3

System requirements

- Mac OS X 10.3, 10.4 or 10.5
- PowerPC or Intel processor
- A VGA graphics card capable of high (16 bit) color and 1024x768 resolution (3D video card recommended)
- 512MB of RAM (1GB recommended)
- A sound card

[Download Alice for Mac OS X 10.4+](#)
[Download Alice for Mac OS X 10.3](#)

Linux

System requirements

- Intel Pentium II or equivalent processor
- A VGA graphics card capable of high (16 bit) color and 1024x768 resolution (3D video card recommended)
- 512MB of RAM (1GB recommended)
- A sound card

[Download Alice for Linux](#)

Mirror/Alternative Download
If the above downloads do not work, please try downloading from our Google Code [mirror](#).

Alice 2.0 vs. Storytelling Alice
What is the difference between Alice 2.0 and Storytelling Alice, and which is right for me?

Alice 2.0 is designed for high school and college students. Storytelling Alice is designed for middle-school students. Storytelling Alice was created by Caitlin Kelleher as part of her doctoral work in Computer Science at Carnegie Mellon University. You can download Storytelling Alice [here](#). For details on the design, development and evaluation of Storytelling Alice, please see [Caitlin's homepage](#).

Figura 2: Escolha do sistema operacional.

File Download

Do you want to open or save this file?

 Name: Alice.zip
Type: Filzip file, 115MB
From: www.alice.org

Open Save Cancel

Always ask before opening this type of file

 While files from the Internet can be useful, some files can potentially harm your computer. If you do not trust the source, do not open or save this file. [What's the risk?](#)

Figura 3: Tela de download.

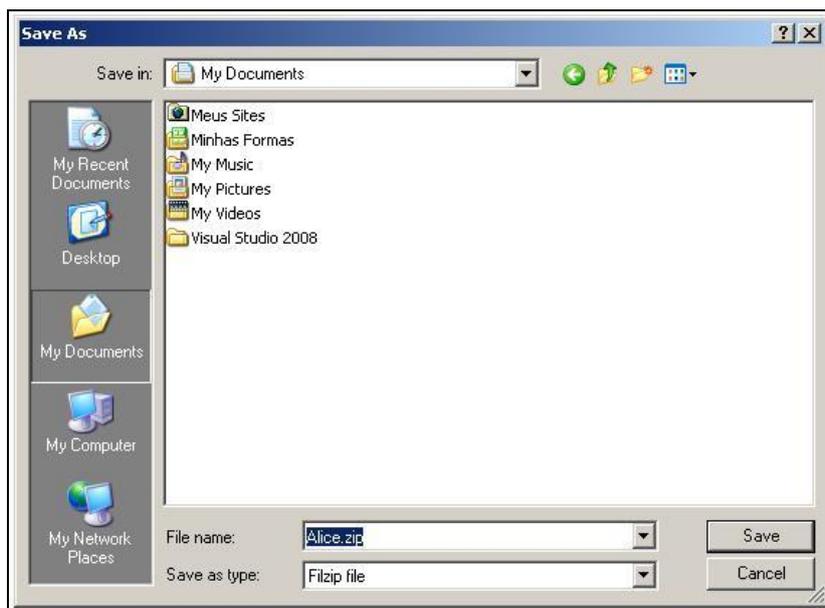


Figura 4: Salvando o arquivo no seu computador.

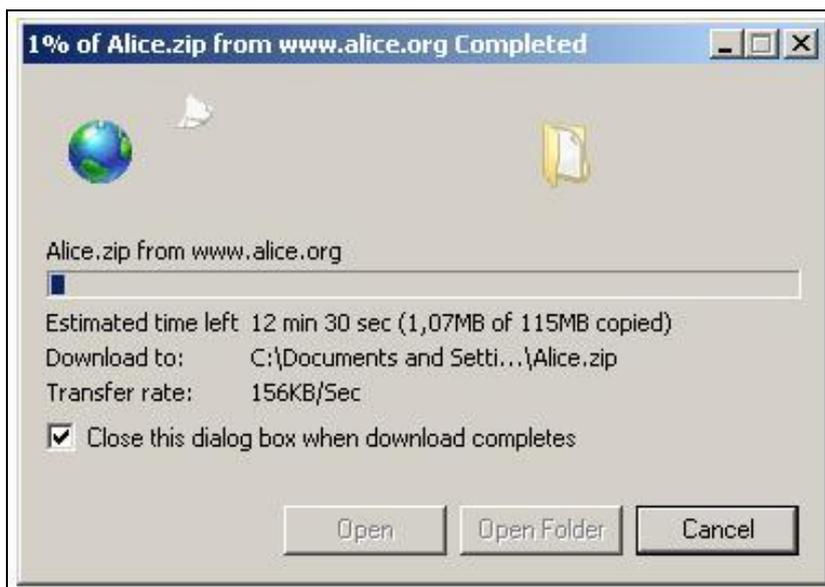


Figura 5: Estatísticas do download.

Depois de concluído o download, descompacte o arquivo zipado no diretório onde o mesmo foi salvo. No diretório principal do software, execute o arquivo Alice.exe.

4. Introdução ao ambiente Alice 2.0

Ao abrir o programa, deverá aparecer uma tela como a mostrada a seguir enquanto o mesmo carrega. A tela seguinte apresenta tutoriais para auxiliar na utilização deste software e para administrar animações criadas anteriormente.

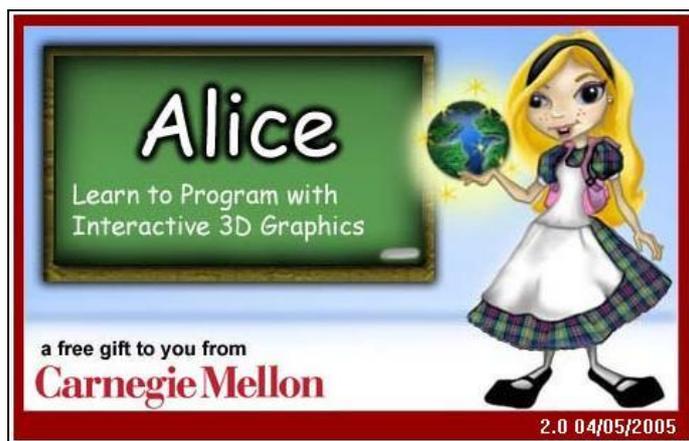


Figura 6: Tela de abertura do ambiente Alice.

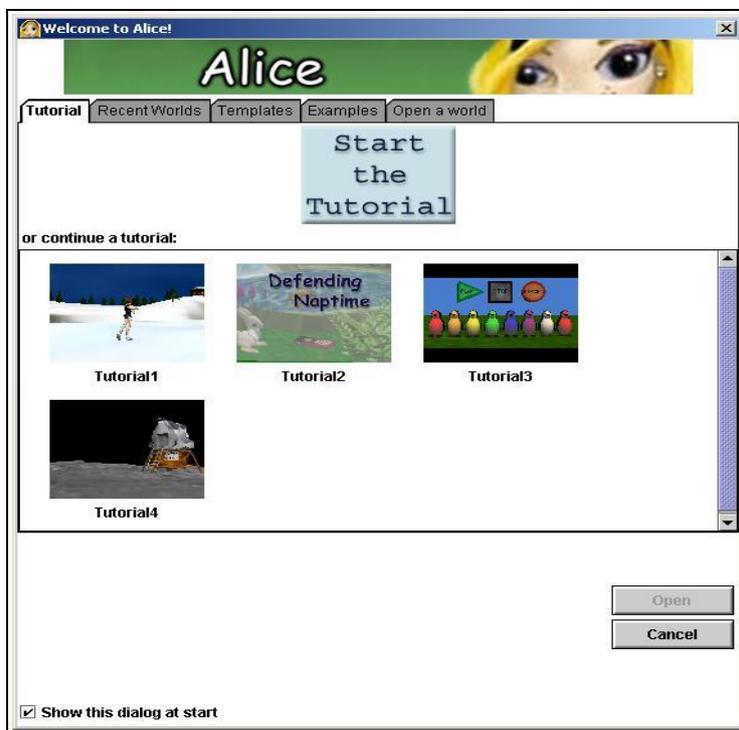


Figura 7: Tela de Boas Vindas.

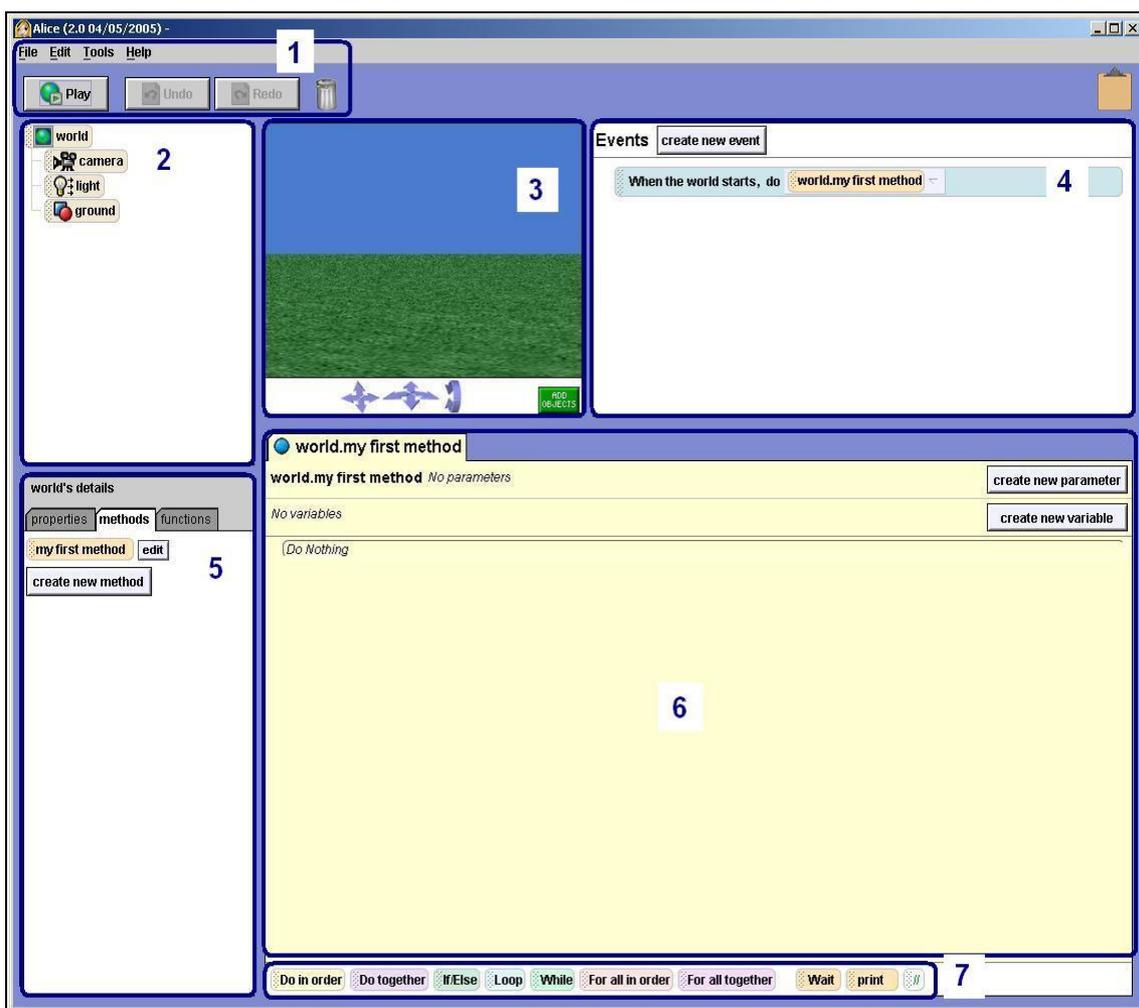


Figura 8: Tela principal do ambiente Alice e suas subdivisões.

A tela principal do ambiente Alice é dividida em 7 partes que serão detalhadas a seguir:

1. Ferramentas, no topo da janela;
2. Lista de Objetos, à esquerda;
3. Mundo ou Cenário, no topo ao centro da tela;
4. Área de Eventos, à direita;
5. Área de Detalhes, à esquerda;
6. Área de Edição, à direita da Área de Detalhes;
7. Funções Lógicas.

A região identificada pelo número 1 compreende as Ferramentas do sistema para salvar e abrir as animações. Nessa região, também existe os botões Play, Undo, Redo e a Lixeira. A Lista de Objetos é identificada pelo número 2 e

apresentará todos os objetos inseridos no Mundo (World), região 3. O cenário começa com 3 objetos básicos, os quais são a Câmera, a Luz e o Terreno, pertencentes ao objeto principal chamado World. Na região 3, Mundo ou Cenário, é o local onde as animações acontecem. É composto pelas setas de navegação para deslocar-se pelo mundo e pelo botão Add Objects. Clicando no botão Add Objects, aparecerá uma tela com mais detalhes como segue abaixo:

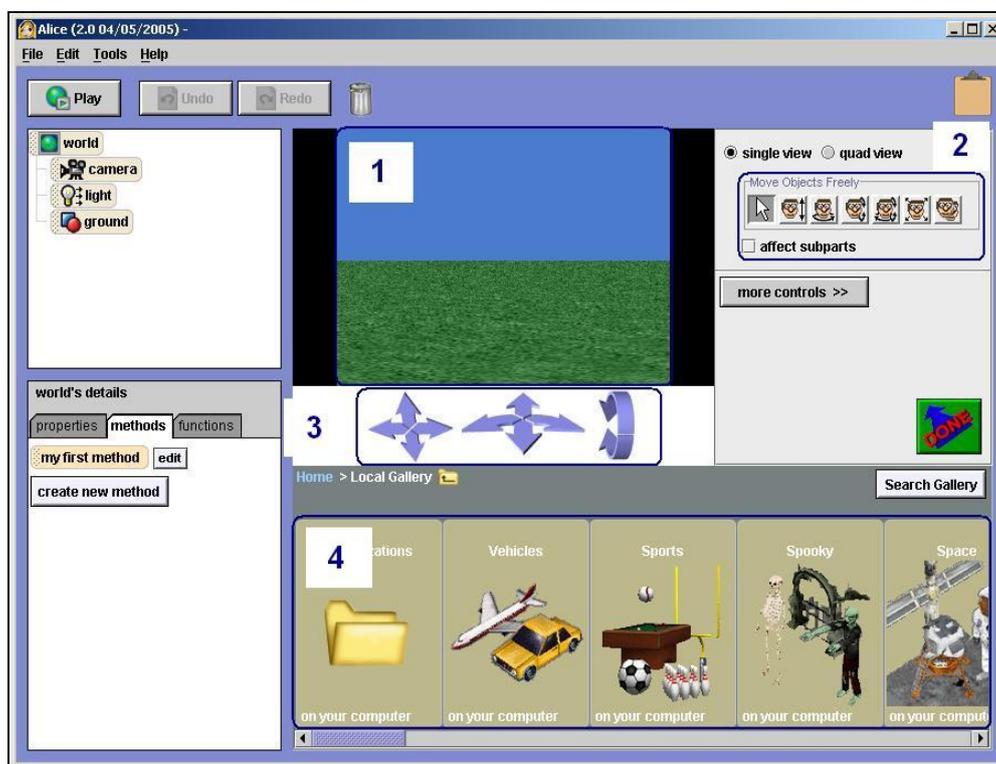


Figura 9: Detalhes do Mundo.

1. Mundo ou Cenário;
2. Propriedades dos objetos;
3. Setas de navegação;
4. Galeria de objetos.

4.1. Objetos

Objetos, em programação, são representações físicas ou não com limites bem definidos e que podem realizar ou sofrer ações através dos métodos no



ambiente em que está inserido. No Alice, alguns objetos possuem o sinal “+” ao lado esquerdo de seu nome, o que indica que tais objetos são compostos por outras partes, também denominadas objetos. Clicando sobre cada um dos objetos, na área abaixo da lista, Área de Detalhes, região 4, irão aparecer os principais métodos do objeto selecionado. Também na Área de Detalhes, existe a aba Propriedades (Properties). Outras propriedades como tamanho e para deslocar os objetos são encontradas no Mundo no botão Add Object.

4.2. Métodos

Cada objeto possui 21 métodos e são os mesmos para todos os objetos do ambiente. São os métodos fundamentais e não podem ser alterados. É com esses métodos que o programador poderá juntá-los de diferentes formas. É através da junção dos métodos fundamentais que poderão ser criados novos métodos. Esses métodos criam as animações que podem ser visualizadas no Mundo. Basta clicar sobre elas e arrastá-las para a região à direita identificada pelo número 6, Área de Edição. Métodos também podem ser eventos, se arrastados para a região 4, Área de Eventos. Através de um evento, pode-se criar movimentos contínuos ou limitados ao andamento da animação. Também é possível criar interações com as animações através de periféricos do computador como o mouse e o teclado, por exemplo. Os eventos serão melhor explicados no final desta apostila.

4.3. Área de Edição e Funções Lógicas

A região de principal interesse é a marcada pelo número 6, Área de Edição, onde é possível criar parâmetros e variáveis através dos respectivos botões. No rodapé dessa região, encontram-se as Funções Lógicas. As principais funções lógicas são *Do in order*, *Do together*, *If/Else*, *Loop*, *While*, *For all in order*, *For all together* e as auxiliares *Wait* e *Print*. As funções e os métodos são arrastados para a Área de Edição.



5. Programando com Alice

Inicialmente, vamos praticar os conceitos vistos até aqui. Os exemplos a seguir apresentam os principais comandos do software. Quanto às funções do ambiente, é necessário que os métodos estejam dentro delas para que sejam executadas. Na utilização de mais de um método, verifique se existem outros métodos ou eventos associados entre si quando for fazer a visualização da animação. É aconselhado um pequeno esboço antes de iniciar a programação propriamente dita, pois, dessa forma, evitará problemas futuros.

Exemplo 1:

- Adicione um objeto no Mundo, por exemplo, um coelho;
- Após, na Lista de Objetos, clique sobre o nome do objeto, nesse caso “Bunny”, para visualizar na Área de Detalhes o que ele sabe fazer;
- Escolha o método “Bunny move” e arraste-o para a Área de Edição;
- Selecione a opção “forward” (mover para frente) e a distância de 5 metros;
- Clique em Play para ver o que ocorre;
- A seguir, adicione um novo método junto ao método anterior. Clique em “Bunny move” e arraste para a Área de Edição logo abaixo do primeiro método;
- Selecione a opção “up” (mover para cima) e a altura de 1 metro;
- Clique em Play para ver o que ocorre;
- Inverta a ordem dos métodos;
- Clique em Play para ver o que ocorre.

No final, salve essa animação com o nome Exemplo1. Você deve ter conseguido algo parecido com o apresentado nas telas que seguem.

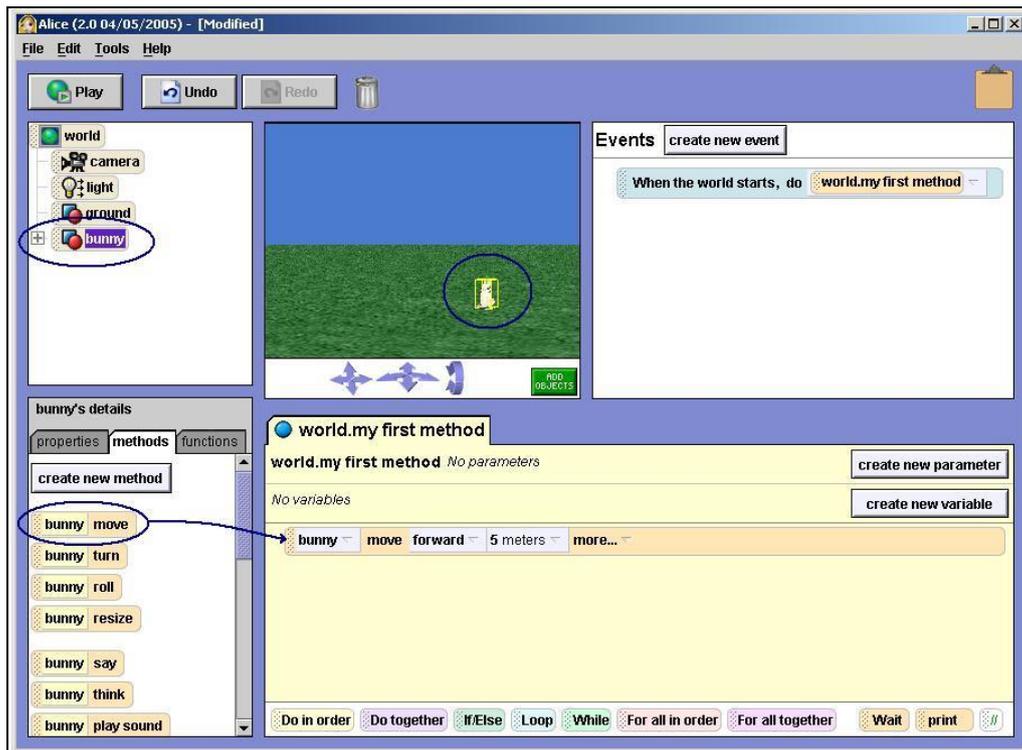


Figura 9: Objeto e método do Exemplo1.

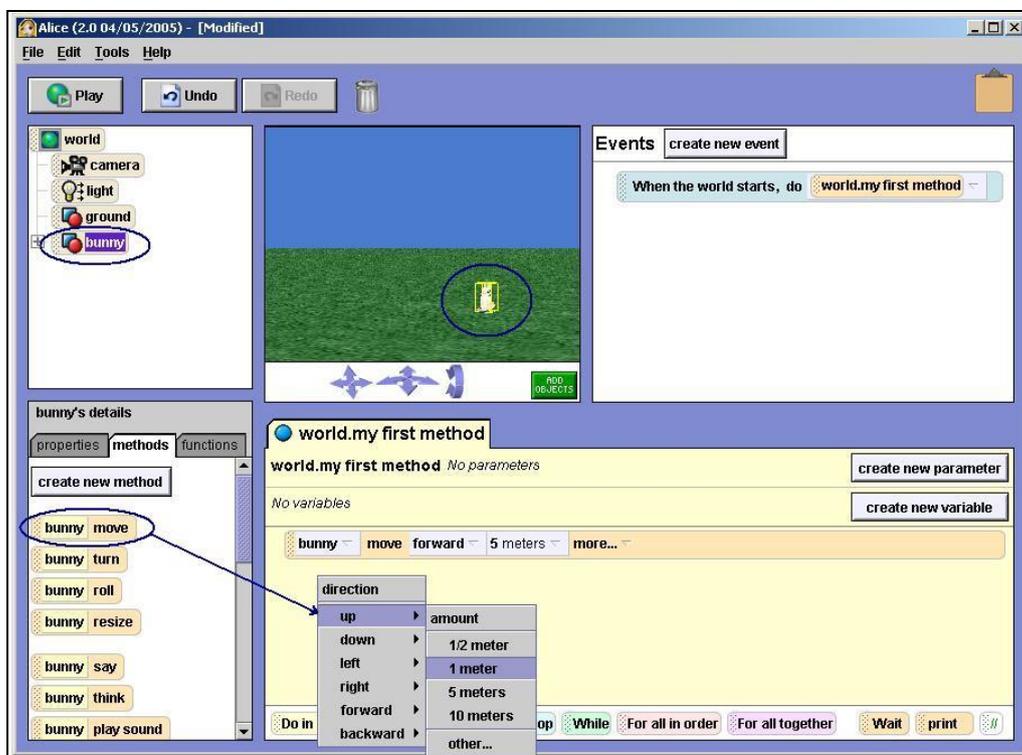


Figura 10: Adicionando um novo método.

Exemplo 2:

- Adicione um objeto com subpartes, por exemplo, um carro;
- Na Área de Edição, adicione a função Loop e escolha “5 vezes” e, dentro dessa, adicione a função *Do together*;
- No objeto “Car” adicionado, clique no sinal de “+” ao lado esquerdo de seu nome para visualizar suas partes;
- Cada parte corresponde a uma das 4 rodas que o veículo possui. Uma a uma, usando o método “turn at speed”, arraste para a Área de Edição dentro da função *Do together*;
- Para cada uma delas, escolha a opção “forward” (virar para frente) com 1 revolução por segundo;
- Clique em Play para assistir a animação.

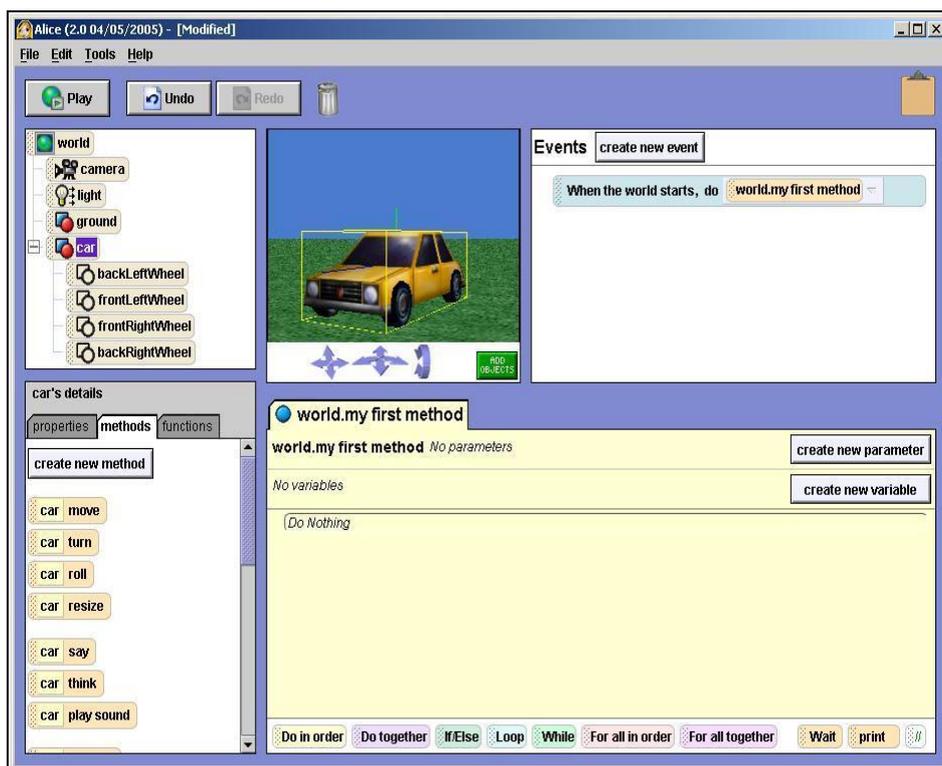


Figura 11: Objeto do Exemplo2.

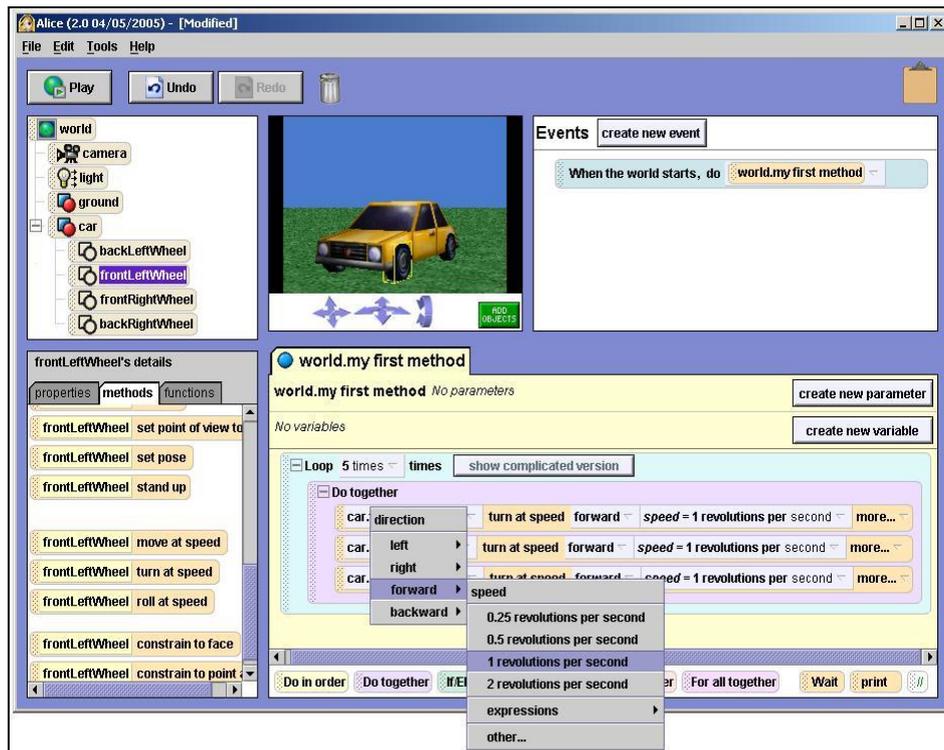


Figura 12: Métodos do Exemplo2.

5.1. Do in order

É a função que determina a realização seqüencial dos métodos. Porém, o ambiente Alice, por padrão, já executa os comandos dessa forma.

Métodos seqüenciais são métodos em etapas. Assim, o método seguinte somente será executado quando o método atual concluir sua execução. Sua utilidade surge quando estamos usando funções que não são executadas seqüencialmente e, dentro dessas, usamos a função *Do in order*. Para utilizar essa função, basta clicar com o botão esquerdo do mouse sobre a função e arrastá-la para a Área de Edição. Os métodos devem estar dentro da função para que os mesmos sejam executados em ordem seqüencial.

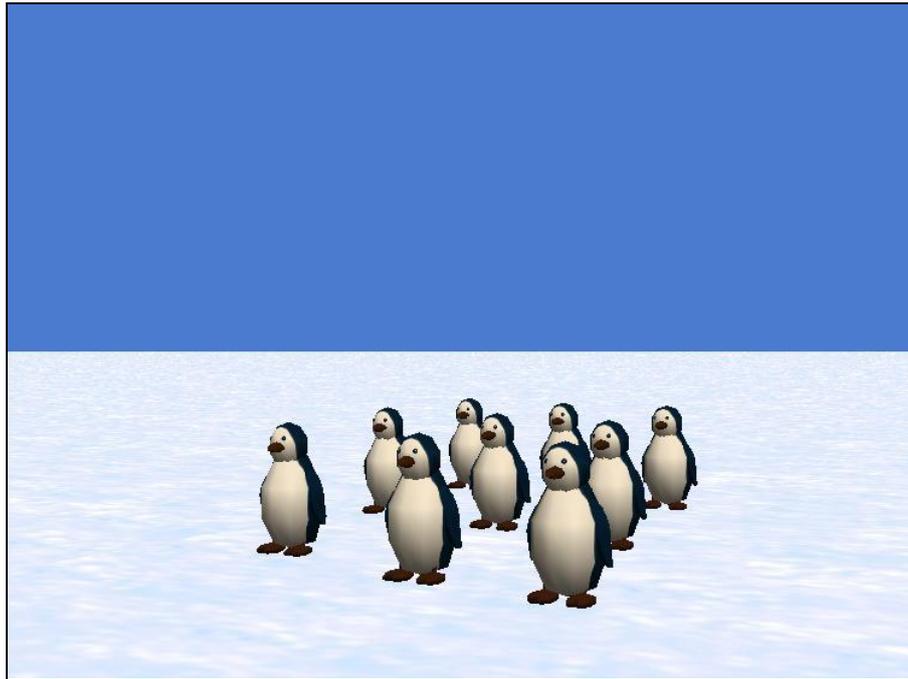


Figura 13: Exemplo a Marcha dos Pingüins.

Nesse exemplo, existem 9 pingüins dispersos por um cenário de neve. Eles não seguem nenhuma ordenação. A partir da função *Do in order*, cada pingüim irá se mover de forma independente, um após o outro.



Figura 14: Métodos do exemplo a Marcha dos Pingüins.

Exercício 1: Crie uma animação no Alice utilizando a função *Do in order*. A animação deverá conter pelo menos 5 objetos.

5.2. *Do together*

É a função essencial para movimentos em conjunto. Todos os métodos que estiverem dentro da função *Do together* serão realizados ao mesmo tempo. É bom observar que métodos diferentes possuem durações diferentes. Para utilização, clicamos com o botão esquerdo do mouse sobre a função e a movemos para a Área de Edição.



Figura 15: Exemplo Autoestrada.

Nesse exemplo, os objetos móveis correspondem aos 4 veículos na estrada. Pela orientação da rodovia, pode-se observar que há dois sentidos possíveis que são seguidos pelos carros. Durante um intervalo de tempo, os veículos se movem de acordo com seus respectivos sentidos ao mesmo tempo.



Figura 16: Métodos do exemplo Autoestrada.

Exercício 2: Crie uma animação no Alice utilizando a função *Do together*. Por se tratar de uma função que envolve mais de um elemento, procure usar vários objetos.

5.3. *If/Else*

A função *If/Else* é certamente uma das funções mais essenciais para a elaboração de uma ação em programação. Essa função necessita de dois resultados, porém, isso não é obrigatório. Para uma expressão arbitrária, dizemos que, se a função *If* possuir o resultado afirmativo, os métodos existentes em seu interior serão executados, caso contrário, serão executados os métodos existentes no interior da função *Else*. Com o botão esquerdo do mouse, clicamos sobre a função e a arrastamos para a Área de Edição.



Figura 17: Exemplo Futebol.

Através de uma variável auxiliar, chamada *distancia*, tem-se o controle da distância entre a bola e a goleira. A função *If* verifica a expressão: $distancia > 4$? Se a expressão for verdadeira, é usada a função *print* para escrever um resultado na tela, senão é escrito outro resultado.

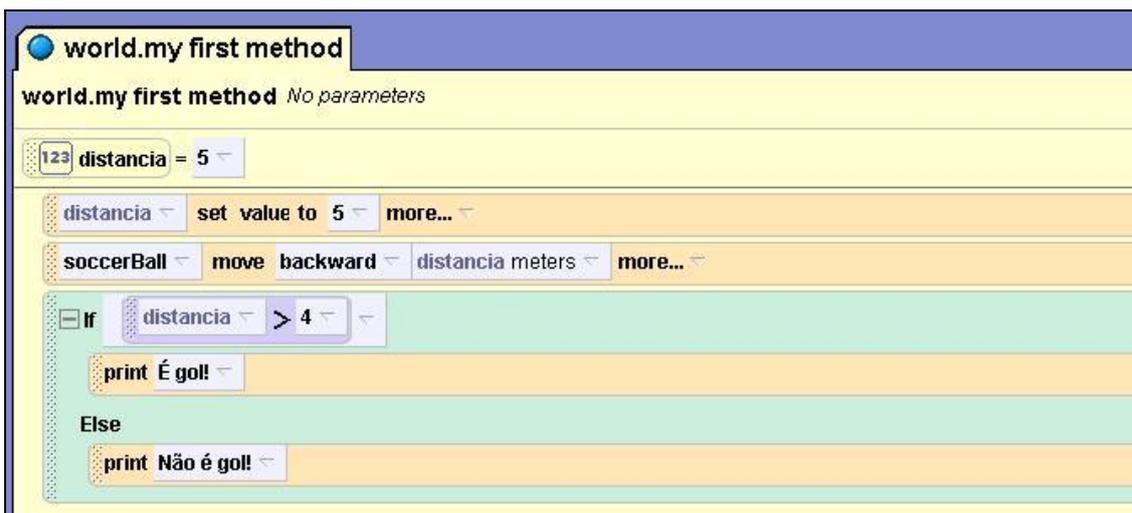


Figura 18: Métodos do exemplo Futebol.

Exercício 3: Crie uma animação no Alice utilizando a função *If/Else*. Pense em uma situação condicional da qual possa ser obtido um valor para sim e outro para não.

5.4. Loop

A função *Loop* determina a quantidade de vezes que uma ação, método ou função vai se repetir. As repetições somente podem ser números naturais até o infinito. A função *Loop* é usada para movimentos contínuos, assim, não é necessário repetir o mesmo código várias vezes. O *Loop* repetirá a ação tantas vezes forem necessárias. Para utilização, a função deve ser arrastada para a Área de Edição como as outras.



Figura 19: Exemplo Fazenda.

Nesse exemplo, foram inseridos alguns objetos com movimentos simples. A animação consiste na câmera ir acompanhando os movimentos dos demais objetos. A animação irá se repetir, 5 vezes para ser exato, conforme a utilização da função *Loop*. Para isso, basta mover os métodos desejados para dentro dessa

função. A seguir, é apresentado como os métodos desse exemplo foram organizados.

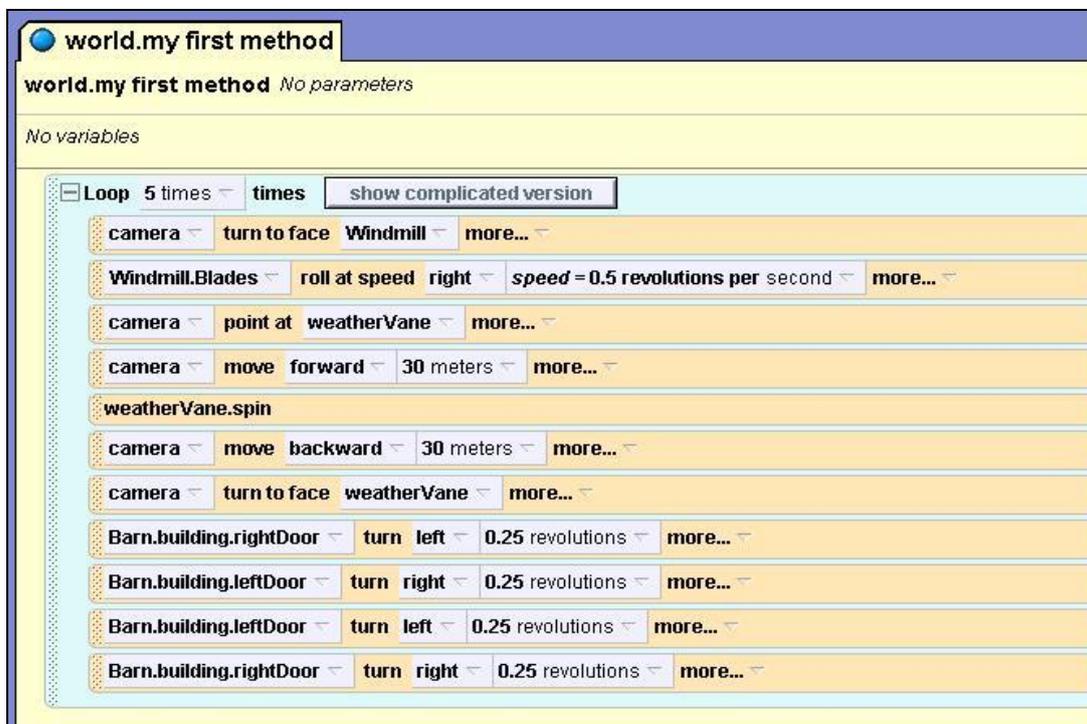


Figura 20: Métodos do exemplo Fazenda.

Exercício 4: Crie uma animação no Alice utilizando a função *Loop*. A repetição pode ser uma ação ou uma pequena história. Escolha a quantidade de vezes para a repetição e verifique se a exibição da animação está como foi realmente planejada.

5.5. While

A função *While* é bem parecida com a função *Loop*. A diferença está no fato da repetição ser condicional, isto é, a função realizará a avaliação de uma expressão antes de iniciar sua execução. A função *While* pode ser descrita como a junção da função *Loop* com a função *If/Else*. Se a condição for verdadeira, a função *While* é executada, senão, avança para o método imediatamente abaixo. Somente através de outra instrução é possível parar a execução da função *While*. Após arrastar essa função para a Área de Edição, os métodos devem ser inseridos dentro da mesma.

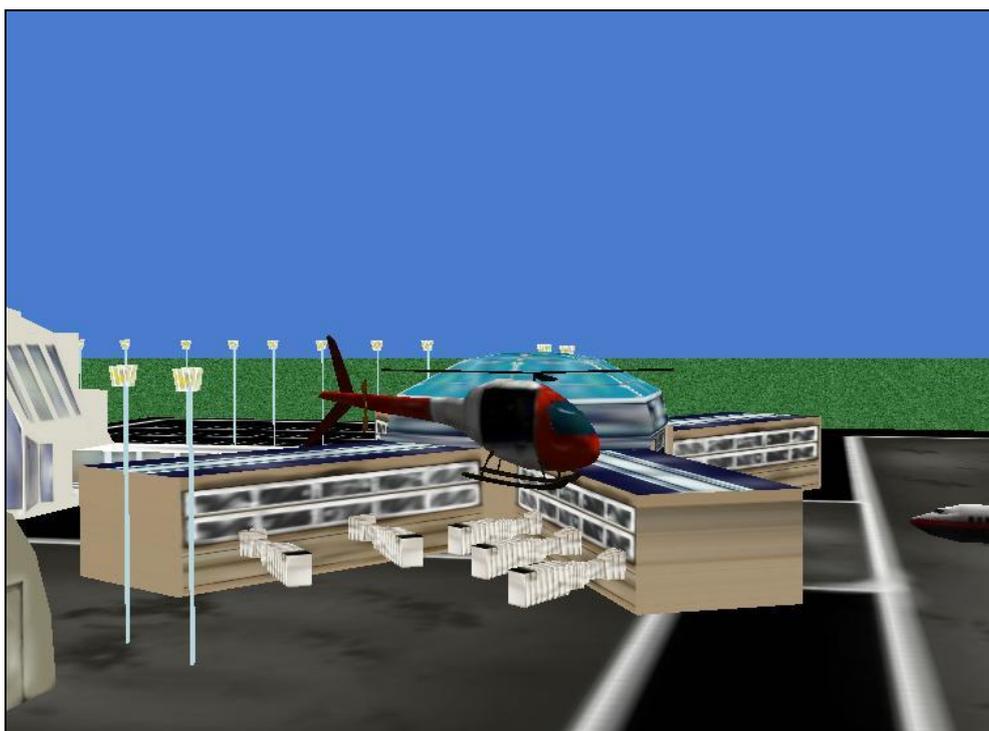


Figura 21: Exemplo Aeroporto.

Nesse exemplo, a função seria repetida por tempo indeterminado, porém, foi condicionado o pouso do helicóptero a uma distância limite com o chão. Dessa forma, quando a condição não for mais satisfeita, a execução irá cessar imediatamente.

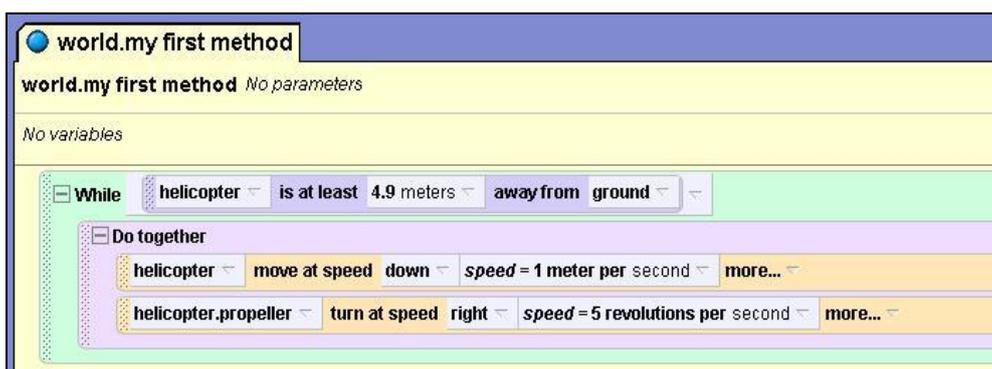


Figura 22: Métodos do Exemplo Aeroporto.

Exercício 5: Crie uma animação no Alice utilizando a função *While*. Pense que primeiro deve ser elaborado uma condição a ser satisfeita. Será através dessa condição que se terá um controle da função *While*. Também pode ser feito com uma variável de controle.

5.6. *For all in order*

É uma das funções para a manipulação de listas. Nessa função, somente será possível modificar ou realizar uma ação em um objeto de cada vez. Ao criar uma lista, você adiciona os objetos na mesma e qualquer comando que envolva a lista se refletirá nos objetos, dependendo do que a lista é composta, se é de objetos ou partes de objetos. Essa função avança de um em um objeto na ordem que foram adicionados na lista criada. Caso não tenha criado uma lista, a inserção da função proporcionará a criação de uma. Basta clicar sobre ela e arrastá-la para a Área de Edição. Ao fazer isso, será solicitada a adição de uma lista ou aparecerá a opção para criar uma.



a 23: Exemplo Casa de Natal.

Nesse exemplo, foi inserida uma sala em específico com dezenas de objetos sejam eles repetidos ou não. Os objetos foram ocultados e através do comando de lista *For all in order* irão aparecer aos poucos.



Figura 24: Métodos do exemplo Casa de Natal.

Exercício 6: Crie uma animação no Alice utilizando a função *For all in order*. Pense em uma situação condicional da qual possa ser obtido um valor para sim e outro para não.

5.7. For all together

É uma lista para mover todos de uma só vez. Ao criar uma lista, você adiciona os objetos na mesma e os comandos afetam todos os objetos ao mesmo tempo e de uma única vez. A forma de adicionar a função e de criar as listas é idêntica a da função anterior.

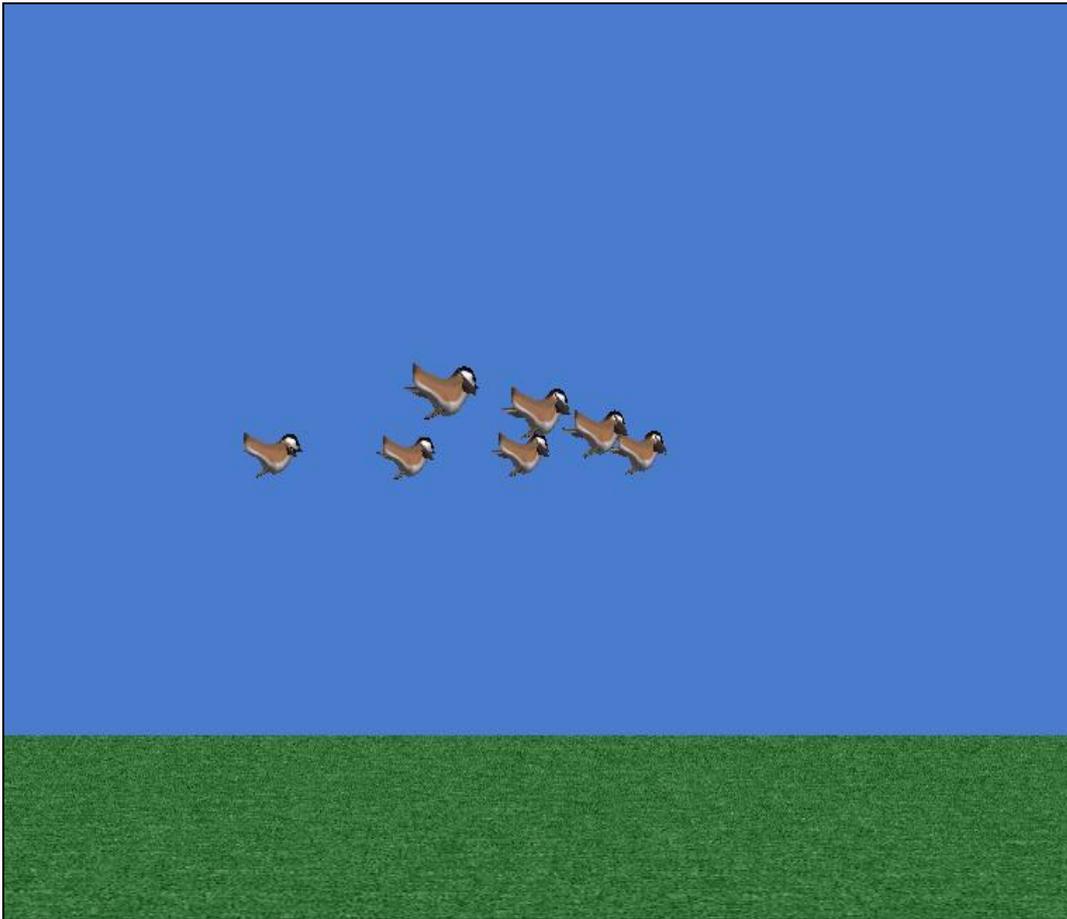


Figura 25: Exemplo Vôo dos Pássaros.

Nesse exemplo, todos os pássaros movem-se ao mesmo tempo. Foi usado mais de uma lista para poder realizar os movimentos das asas dos pássaros. Os movimentos são simultâneos durante todo o vôo.

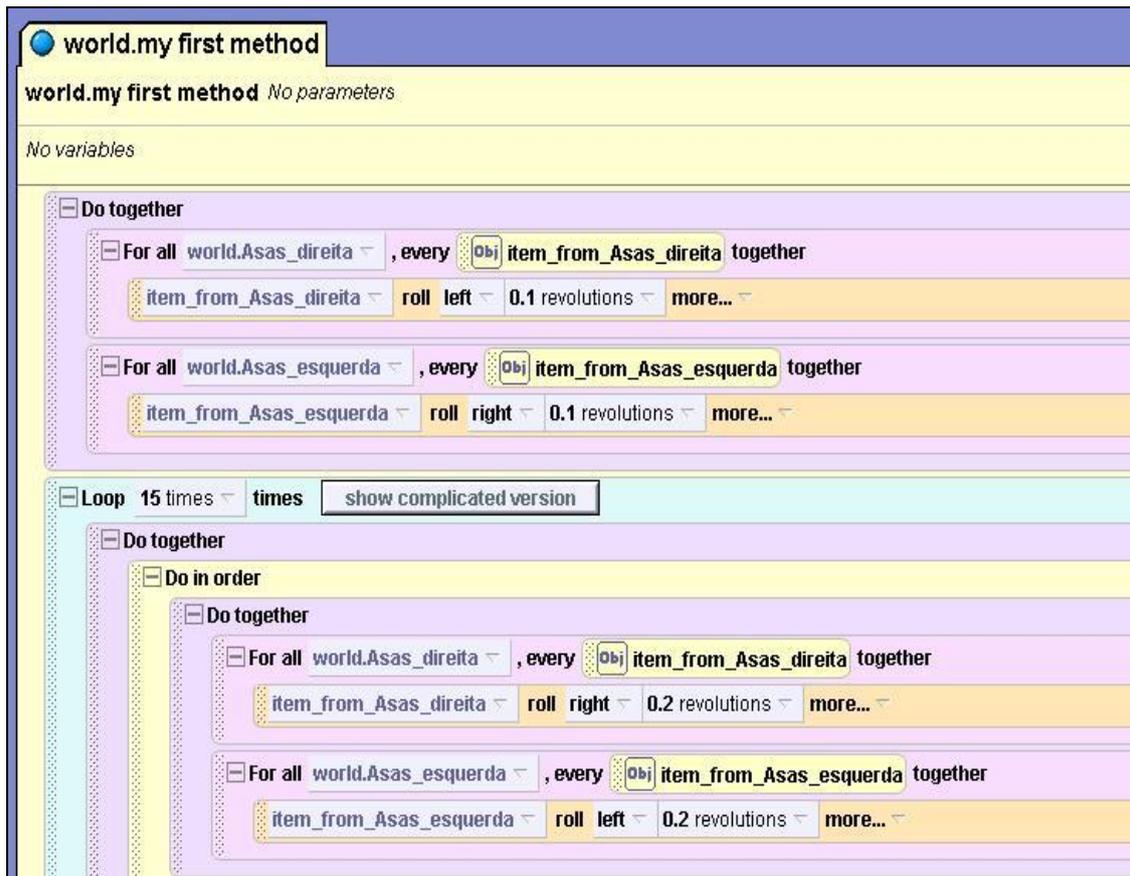


Figura 25.1: Métodos do exemplo Vôo dos Pássaros.

Exercício 7: Crie uma animação no Alice utilizando a função *For all together*. Pense em uma situação condicional da qual possa ser obtido um valor para sim e outro para não.

6. Eventos

Os Eventos que podem ser criados no ambiente Alice podem ser métodos, variáveis ou algum comando que tenha uma reação através do teclado ou mouse. Os eventos constituem ações ou pedaços de animações que irão acontecer no decorrer da história ou animação criada. É comumente usado para sons e para interação. A seguir tem-se uma visão geral da tela de eventos do ambiente:

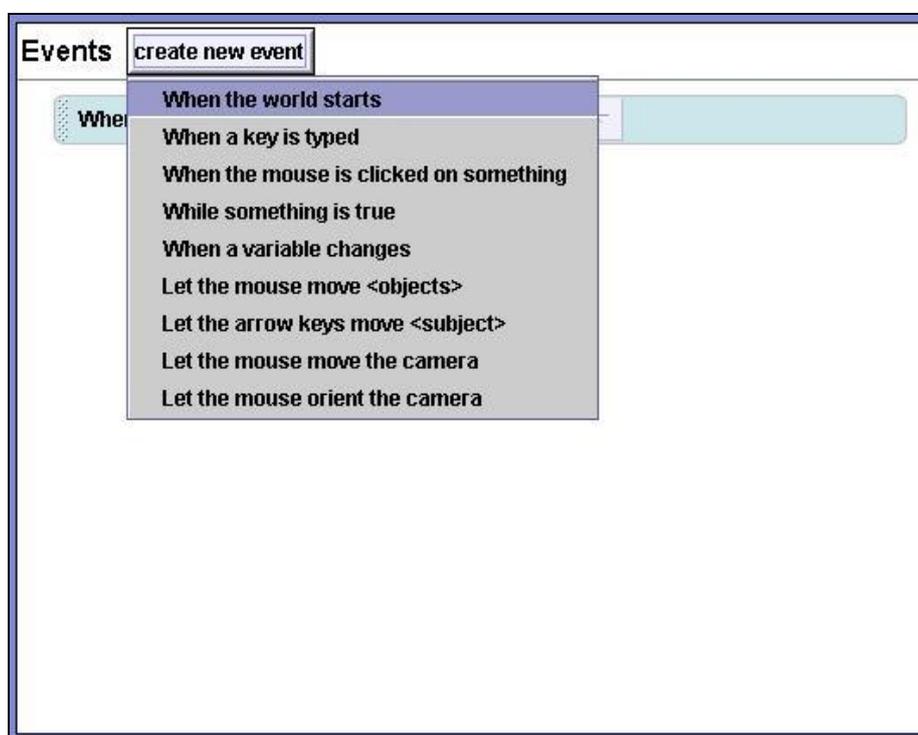


Figura 26: Área de Eventos.

Os eventos podem ser Não-Interativos ou Interativos. Além disso, possuem 3 tipos: *When*, *While* e *Let*. Os eventos do tipo *When* podem ser alterados em *While*, com exceção do evento *When a variable changes*, e os do tipo *Let* não podem ser alterados. Por padrão, o evento inicial é *When the world starts*, o qual chama o primeiro método para execução.



6.1. Eventos Não-Interativos

Os eventos não-interativos possuem uma execução breve ou contínua. O fato de não serem interativos é que não utilizam periféricos para ocorrerem. Normalmente são métodos, funções ou objetos que estão inseridos no evento. São eventos Não Interativos: *When the world starts*, *While something is true* e *When a variable changes*.

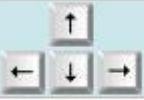


6.2. Eventos Interativos

São Eventos Interativos: *When a key is typed*, *When the mouse is clicked on something*, *Let the mouse move <object>*, *Let the arrow key move <subject>*, *Let the mouse move the camera*, *Let the mouse orient the camera*.



Events

Let  move camera

Let  move the camera

When is typed, do

When is typed, do

When is typed, do

Events

While is pressed

 Begin:

 During:

 End:

When is typed, do

When is typed, do



Trabalho Final

O trabalho final consiste em elaborar uma pequena história usando todos os recursos abordados nessa apostila. Não há limites para sua criatividade. Devem existir novos métodos a partir dos já existentes, utilização de todas as funções. Os seguintes passos possuem o objetivo de auxiliá-lo(a) na construção desse trabalho.

- 1º. Pense na história a ser elaborada, pode ser uma já existente;
- 2º. Monte esboços antes de iniciar a construção da história. Fatos que irão ocorrer ou como as personagens irão interagir. Tenha em mente o tipo de história e as personagens que vão fazer parte dela;
- 3º. Inicie a animação construindo um cenário que já deve ter sido especificado anteriormente;
- 4º. Lembre-se de usar as capturas de poses logo no início se for utilizá-las;
- 5º. Efeitos de câmeras e luz são bem-vindos;

Bom trabalho!

Apêndice A

Propriedades

Os objetos possuem amplas características e a maioria delas são modificáveis. Tais características encontram-se na Área de Detalhes tendo uma de suas 3 abas, a aba Properties (Propriedades). A seguir, serão definidas as principais propriedades encontradas na maioria dos objetos.



Figura 27: Propriedades da maioria dos objetos.



Create New Variable: ao contrário do botão existente na Área de Edição, esse permite a criação de uma variável global que poderá ser usada em diferentes métodos.

Capture Pose: é a principal propriedade quando se trata da movimentação de partes dos objetos. Com ela, pode-se capturar, isto é, memorizar a pose atual do objeto e utilizá-la em outros momentos quando desejado. Mais de uma pose pode ser capturada e elas podem ser utilizadas em outros objetos diferentes daqueles que tiveram suas poses capturadas. Essas poses funcionam apenas nas partes do objeto principal e não servem para movimentações pelo Mundo;

Color: é possível alterar a cor principal do objeto. Clicando nessa propriedade, aparecerá uma lista de cores com opções para mudanças de tonalidades;

Opacity: corresponde à transparência do objeto, a opacidade. Um objeto visível possui a opacidade em 100%. Logo, para 0% de opacidade, o objeto se torna invisível;

Vehicle: através dessa propriedade é possível fixar um objeto a outro, isto é, um objeto se tornará dependente de outro. Por exemplo, machado e cavaleiro são objetos diferentes. Porém, posso colocar o machado na mão do cavaleiro e usar esta propriedade para que o mesmo fique na mão desse cavaleiro durante os movimentos de sua mão;

Skin texture: para cada objeto contido no Mundo, o Alice registra o seu tipo de textura nessa opção. Dessa forma, é possível variar os objetos com texturas de outros objetos que já estejam no Mundo;

FillingStyle: serve para mudar a forma de visualização dos objetos. É possível alterar, além da visualização normal, para frames e para pontos;

pointOfView: é possível alterar entre a posição inicial ou outras que o mesmo objeto já usou no Mundo em um momento anterior;

isShowing: é possível alterar a qualquer momento se um objeto estará visível ou não;



Import Sounds: para importar sons nas extensões “.wav” e “.mp3”;

Importar Texturas: para importar novas texturas;

Observação: As opções referente as Seldom Used Properties, como o próprio nome diz, são raramente usadas não possuindo uma importância muito significativa.



Apêndice B

Importando figuras

O ambiente Alice suporta a importação das principais extensões de figuras como gif, jpg, jpeg, tif, png, bmp, tiff. Cada imagem adicionada ao Mundo será interpretada como um objeto e será dotada dos principais métodos como todos os outros objetos existentes. Para inserir as figuras, siga os passos e as telas abaixo:

- Nas Ferramentas do ambiente Alice, clique em File;
- Após, clique na opção Make Billboard;
- Uma nova janela aparecerá. Escolha a imagem desejada no diretório correspondente e clique em “Import”.



Figura 28: Menu File – opção Make Billboard.

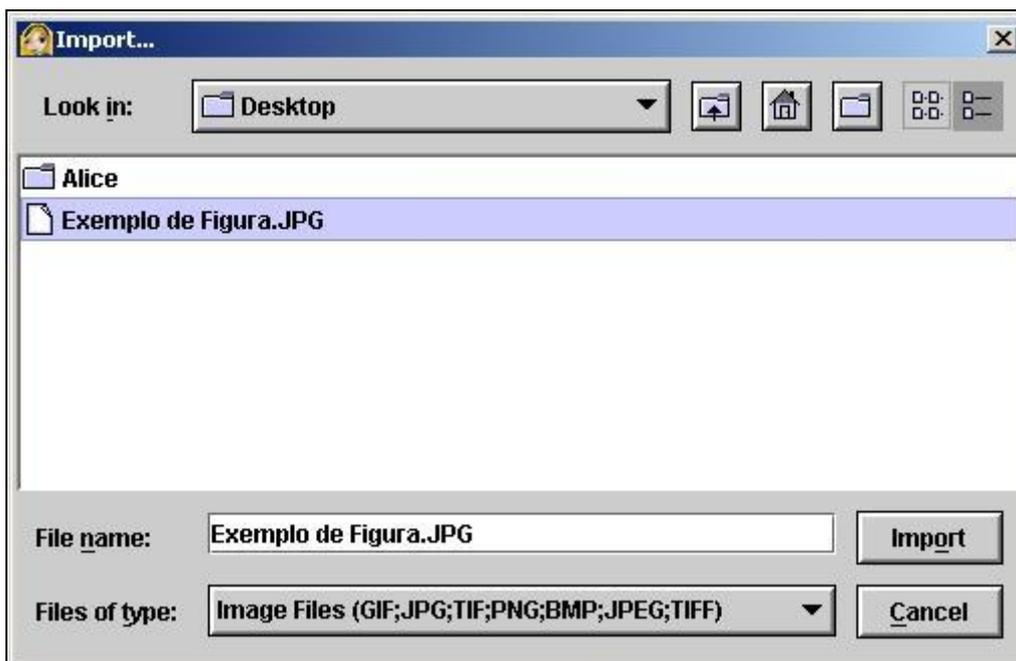


Figura 29: Importando as figuras.

Importando outros objetos

No ambiente Alice, também é possível importar outros elementos 3D provenientes de outros programas de CAD. A extensão que ele aceita é a extensão “.ase” que pode ser obtida a partir do software 3D Max Studio.

Apêndice C

Criando vídeos

Nas Ferramentas do ambiente Alice, no menu File, existe uma opção chamada Export Movie, porém ela não funciona na versão Alice 2.0. Ao clicar nessa opção, irá aparecer uma mensagem dizendo que tal função ainda não foi implementada. Então, dirija-se ao site www.alice.org e faça o download da versão Alice 2.2. Dessa forma, ao clicar na opção Export Video, aparecerá a tela a seguir:

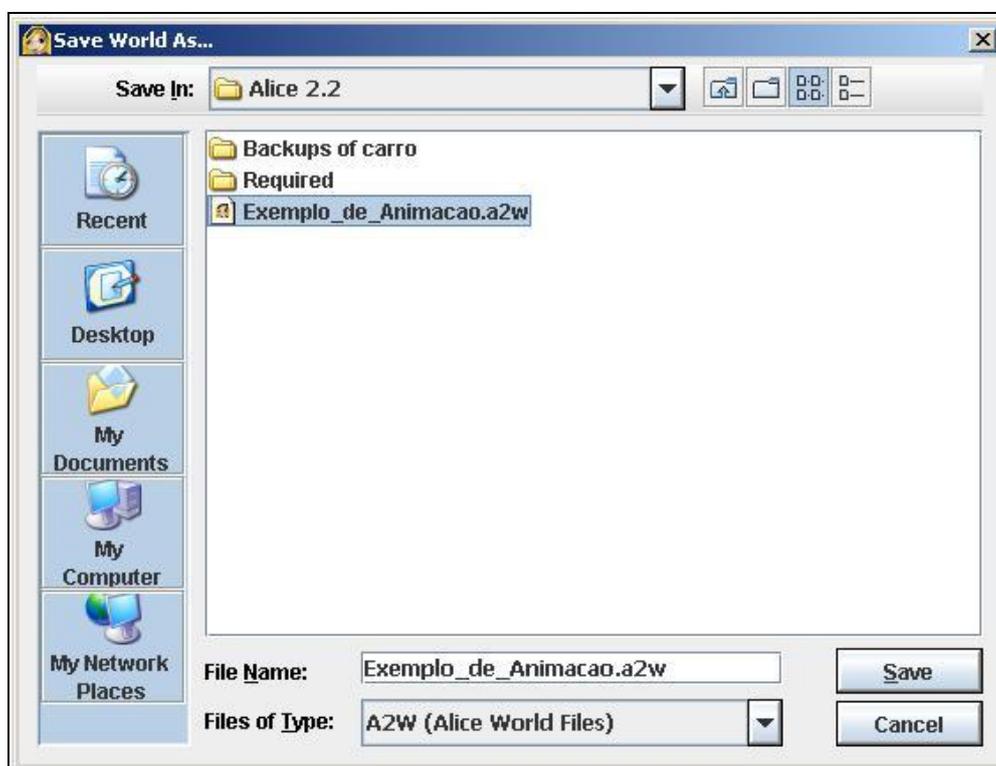


Figura 30: Opção Export Video.

Selecione o arquivo com a extensão do Alice para fazer o vídeo e em seguida clique no botão Save.

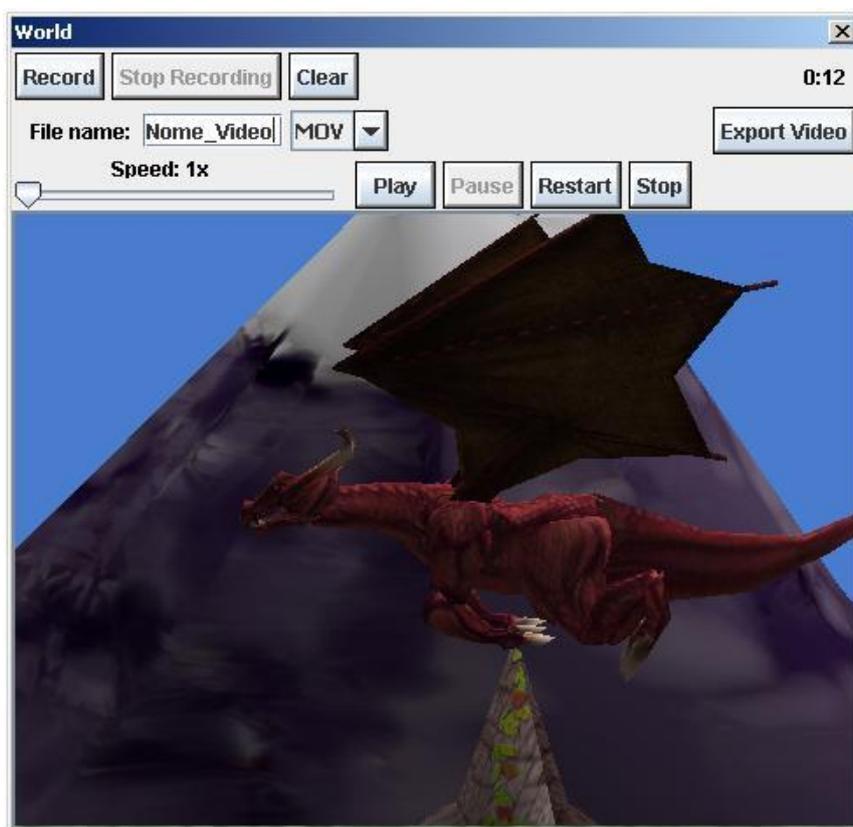


Figura 31: Tela de Gravação.

Na Tela de Gravação, o vídeo será criado a partir da reprodução da animação. Ela reproduzirá exatamente a mesma animação quando clicamos no botão Play na tela principal, porém, nessa tela, deve-se salvar a animação durante sua execução. Inicialmente, clique no botão Record para começar a salvar o vídeo. Caso a animação não inicie automaticamente, clique no botão Play logo após. Quando a animação terminar, clique no botão Stop Recording e em seguida no botão Stop para parar a animação. No campo File name, digite o nome do vídeo que você acabou de criar. Quanto as extensões do vídeo, por enquanto, somente é possível salvar na extensão “.mov”, que é reproduzida pelo Quick Time, por exemplo. Para finalizar, clique na opção Export Video localizada à direita. Aparecerá uma nova tela pedindo que você escolha o diretório onde deseja salvar o vídeo que acabou de criar. É comum que o vídeo possua sucessivas trancadas durante sua execução, pois esse bug ainda não foi corrigido.



Referências Bibliográficas

Site Alice.org. Disponível em: <http://www.alice.org>. Acesso em 11 jun. 2009.

Cooper, Stephen; Dann, Wanda; Pausch, Randy. Teaching Objects-first In Introductory Computer Science SIGCSE 2003. 5 p.

Conway, Mathew J. Alice: Easy-to-Learn 3D Scripting for Novices. Dissertação de PhD, 1997. Estados Unidos. 242 p.

Dann, Wanda P.; Cooper, Stephen; Pausch, Randy. Learning to Program with Alice. Estados Unidos: Prentice Hall, 2005.375 p.