

UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**PROGRAMAÇÃO CRIATIVA E NARRATIVAS INTERATIVAS
A BIBLIOTECA NINE.JS**

NATALIA PINHEIRO DA SILVA

VITÓRIA DA CONQUISTA - BA
NOVEMBRO/2021

**PROGRAMAÇÃO CRIATIVA E NARRATIVAS INTERATIVAS
A BIBLIOTECA NINE.JS**

NATALIA PINHEIRO DA SILVA
GIDEVALDO NOVAIS DOS SANTOS

Projeto entregue à disciplina Trabalho Supervisionado II como requisito parcial para aprovação na disciplina, ministrada pelo Prof. MSc. Gidevaldo Novais dos Santos, no PL 2020.2.

VITÓRIA DA CONQUISTA - BA
NOVEMBRO/2021

DEDICATÓRIA

Dedico esse trabalho a mim mesma, porque foi difícil chegar até aqui.

AGRADECIMENTOS

Primeiramente, agradeço a minha família. Minha irmã Lettycia e meus pais, Milena e Edson. Não estaria onde estou sem eles, e não seria a pessoa que sou sem eles.

Ao melhor presente que a UESB me deu, meus amigos Jenifer, Vinicius, Victória e Tharcio. Obrigada por estarem sempre comigo.

O meu orientador, Gil, por todo o apoio e por me guiar. A professora Maísa, por acreditar no Nine desde o começo, mesmo quando nem eu mesma tinha certeza do que estava propondo.

Agradeço os professores, Celina, e todos os colegas que me ajudaram e inspiraram todos esses anos.

Agradeço ainda Karine, Manu e Raquel, por estarem sempre presentes apesar da distância.

And I also thank my good friend Quinto.

Leitura e conhecimento são as raízes da sabedoria,
então plante sua árvore.

RESUMO

Contar histórias é uma habilidade natural do ser humano. Desde o início das civilizações, as pessoas contavam histórias para dar contexto a eventos, transmitir ideias e expressar sentimentos. O advento da tecnologia significou uma verdadeira revolução, influenciando em todos os segmentos da vida humana, inclusive o ato de contar histórias. A computação proporcionou às narrativas aspectos de interatividade antes nunca vistos e, até hoje, bastante inexplorados. Embora a programação criativa esteja cada vez mais popular, as ferramentas existentes possuem foco em imagem e áudio, e as poucas que são focadas em texto são anglocêntricas e não padronizadas. Sendo assim, o objetivo desse trabalho foi explorar a biblioteca Nine.js, uma framework de programação criativa que possibilita a escrita de narrativas interativas de forma escalável e em português.

Palavras-chave: Programação criativa; narrativa interativa; narrativa não-linear.

ABSTRACT

Storytelling is a natural human ability. Since the beginning of civilizations, people told stories to give context to events, convey ideas and express feelings. The advent of technology caused a real revolution, influencing all segments of human life, including the act of telling stories. Computing provided narratives with aspects of interactivity that had never been seen before and, until today, have been largely unexplored. While creative programming is increasingly popular, existing tools focus on image and audio, and the few that focus on text are Anglocentric and non-standard. Therefore, the goal of this work was to explore the Nine.js library, a creative programming framework that allows the writing of interactive narratives in a scalable way and in Portuguese.

Keywords: Creative coding; interactive narrative; non-linear narrative.

SUMÁRIO

1 INTRODUÇÃO	9
1.2 PROBLEMA DE PESQUISA	10
1.3 OBJETIVOS	10
1.3.1 Objetivo geral	10
1.3.2 Objetivos específicos	10
1.4 JUSTIFICATIVA	10
1.5 METODOLOGIA	11
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 PROGRAMAÇÃO CRIATIVA	13
2.1.1 Processing	13
2.1.2 p5.js	15
2.2 NARRATIVAS	17
2.2.1 Narrativas interativas	18
2.3 TRABALHOS RELACIONADOS	20
3 NINE.JS	25
3.1 ESTRUTURA DE UM PROGRAMA EM NINE.JS	25
3.2 FUNÇÕES DE CONFIGURAÇÕES	27
3.2.1 Título	27
3.2.2 Autor	27
3.2.3 Cor	27
3.2.4 Cor de fundo	28
3.2.5 Tamanho da fonte	28
3.3 FUNÇÕES DA HISTÓRIA	28
3.3.1 Adicionar texto	28
3.3.2 Adicionar texto	29
3.3.3 Adicionar imagem	29
3.3.4 Adicionar opções	29
3.3.5 Entrada	31
3.4 PERSONAGENS	31
3.5 ESTADO	32
4 CONSIDERAÇÕES FINAIS	34
REFERÊNCIAS	35

1 INTRODUÇÃO

A arte é uma atividade humana ligada a manifestações de ordem estética e comunicativa, podendo ser construída em conjunto com as mais diversas áreas, como arquitetura, culinária, desenho, escrita, música, dança, cinema e até mesmo jogos eletrônicos. A arte está em toda parte, inclusive na programação.

Segundo Cormen, Leiserson e Rivest (2017, p. 17),

Informalmente, um algoritmo é qualquer procedimento computacional bem definido que toma algum valor ou conjunto de valores como entrada e produz algum valor ou conjunto de valores como saída. [...] Também podemos considerar um algoritmo como uma ferramenta para resolver um problema computacional bem especificado. O enunciado do problema especifica em termos gerais a relação desejada entre entrada e saída.

Em se tratando de computação, um campo que aprecia definições formais, soa bastante estranho relacionar essa área a formas de expressão artística. Entretanto, a Terceira Revolução Industrial, também conhecida como Revolução Digital, trouxe uma expansão significativa na maneira como as pessoas usam computadores digitais, essas máquinas deixaram de servir exclusivamente fábricas, instituições militares e acadêmicas, e se tornaram parte do cotidiano das pessoas (DUARTE, 1999). Com isso, houve um aumento expressivo de áreas assistidas ou impactadas pela computação, entre elas, a arte.

Embora a programação e escrita de algoritmos no geral seja muito conhecida como um meio de resolver problemas, como visto anteriormente no conceito dado por Cormen (2017), existe uma subárea chamada programação criativa, ou *creative coding*, que vem ganhando cada vez mais popularidade nos dias atuais. A programação criativa deixa de tratar a tecnologia como uma ferramenta, e refere-se a ela como uma nova forma de expressão.

Apesar de ser uma área relativamente nova e pouco explorada, a programação criativa aplicada no contexto da computação gráfica ou sonora possui destaque considerável, deixando assim muito espaço para a exploração da programação criativa possibilitando a construção de narrativas.

A computação atribui à arte características de imersão, interatividade, ubiquidade, conectividade, entre outras, que são traços bastante particulares e que proporcionam à arte

computacional um aspecto singular. A programação de computadores aliada à escrita de narrativas auxilia, entre outras coisas, no desenvolvimento das chamadas narrativas interativas; uma modalidade do entretenimento digital que permite que o leitor, ou usuário, possa direcionar o desenvolvimento da história com suas decisões. Cabe, portanto, uma investigação a fundo de quais estruturas computacionais e lógicas podem assistir a construção de narrativas interativas e, ainda, de que maneira estas podem ser modeladas como algoritmos.

1.2 PROBLEMA DE PESQUISA

A programação criativa se refere à construção de programas de computador com foco na expressão em vez da funcionalidade. Popularmente associada com processamento de imagem e som, essa modalidade de programação possui inúmeras possibilidades. Existem bastantes ferramentas que possibilitam a escrita de códigos criativos, as quais têm ganhado cada vez mais destaque. Entretanto, pouco é dito sobre a conjunção de programação e a narração de histórias. De que maneira é possível escrever narrativas interativas com código?

1.3 OBJETIVOS

1.3.1 Objetivo geral

Exibir uma forma de escrever narrativas interativas com código.

1.3.2 Objetivos específicos

- Explorar o conceito da programação criativa e suas diferentes facetas;
- Localizar a construção de narrativas na programação criativa;
- Apresentar uma *framework* para a construção de narrativas interativas com código.

1.4 JUSTIFICATIVA

Com o passar do tempo, a tecnologia tem se tornado essencial na vida humana e presente em todas as áreas do conhecimento. É difícil citar um campo que não seja afetado direta ou indiretamente pela computação. Com isso, a programação é uma área que vem ganhando bastante destaque.

É possível definir um algoritmo como “uma ferramenta para resolver um problema computacional bem especificado” (CORMEN, LEISERSON e RIVEST, 2017, p. 17). Essa definição possui um tom bastante formal e concreto, adjetivos esses que são relativamente comuns na área. Entretanto, existe uma área pouco explorada da programação chamada programação criativa, a qual não trata produtos de software como simples meios para resolver problemas, mas como uma forma de expressão.

Na programação criativa, a intenção não é cumprir um determinado conjunto de requisitos. Nela, o objetivo é comunicar uma ideia, um pensamento, causar emoções, provocar admiração, criar uma forma de arte, de certo modo.

A programação criativa pode utilizar os mais diversos materiais, como imagens, sons, movimentos, texto e *hardware*; os dois primeiros sendo os mais comuns. Por esse motivo, existem inúmeras *frameworks* que auxiliam essa modalidade de programação. Contudo, a intersecção entre a programação criativa e a escrita de histórias ainda não possui muito destaque.

Existe uma modalidade de histórias não-lineares chamada narrativas interativas, as quais têm se tornado cada vez mais populares, portanto, atualmente, há algumas linguagens, bibliotecas e *frameworks* focadas no desenvolvimento dessas narrativas. Porém cada uma delas possui uma sintaxe própria e em geral muito diferente umas das outras, então não existe padronização ou consistência nessas ferramentas, além de serem bastante anglocêntricas. Logo, foi identificada a oportunidade de desenvolver uma ferramenta escalável de desenvolver narrativas interativas e, além disso, em português. A ferramenta aqui apresentada, Nine.js, foi desenvolvida durante o período de uma iniciação científica na Universidade Estadual do Sudoeste da Bahia.

1.5 METODOLOGIA

Esse trabalho tem como método de desenvolvimento as seguintes etapas:

- Pesquisa bibliográfica de caráter exploratório como foco principal, a fim de conhecer e compreender os conceitos relevantes sobre programação criativa e narrativas interativas.
- Elaboração de um mapeamento não sistemático de ferramentas que possibilitam a criação de narrativas interativas programaticamente, elencando seus

diferenciais, seus pontos fortes e fracos.

- Com base nas oportunidades encontradas a partir dos passos anteriores, expor uma nova maneira de construir narrativas interativas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 PROGRAMAÇÃO CRIATIVA

Programação criativa, ou *creative coding*, como é mais conhecida, é uma forma de programação que subordina a função à expressão. A programação criativa estende a programação de apenas escrever código para uma atividade artística que inclui código. Tem potencial de envolver elementos físicos, eletrônicos, sensores e interação com o mundo físico (Ackermann, Egger e Scharlach, 2020).

Sobreira, Dos Santos e Sobreira (2020) trazem que, de acordo com o Guia de Computação Criativa da Escola de Pós-Graduação em Educação de Harvard,

A computação criativa propõe uma relação entre a computação e os interesses pessoais de cada indivíduo, baseado na criatividade, imaginação e interesses pessoais. Trabalha a computação como oportunidade de criação, não apenas para consumo de tecnologias prontas, favorecendo assim, o desenvolvimento de práticas para a criação de meios computacionais interativos que podem ser utilizados no cotidiano (SOBREIRA, DOS SANTOS E SOBREIRA, 2020, p. 30).

Segundo González, Preciado e Terrazas (2015), a programação criativa surgiu sob a influência da arte generativa, uma forma de arte que foi criada completa ou parcialmente por um sistema autônomo (MCCORMACK et al, 2014), a qual atraiu artistas, desenhistas, arquitetos, músicos e até mesmo escritores interessados em estender e aplicar suas habilidades criativas.

Em geral, a programação criativa inclui uma ampla coleção de ferramentas, *frameworks* de programação e *software* — Processing¹, p5.js², assim por diante —, bem como vários contextos de criação de arte — *code poetry*, *live coding*, cultura *hacker*, cultura *maker*, *software* livre, entre outros (DUFVA, 2021).

2.1.1 Processing

Certamente a mais popular das *frameworks* voltadas para a programação criativa, Processing surgiu do grupo de pesquisa Aesthetics + Computation do Media Lab do Instituto de Tecnologia de Massachusetts — MIT (GREENBERG, 2007).

Processing é um software de rascunhos flexível e uma linguagem para aprender a codificar dentro do contexto das artes visuais. Desde 2001, o Processing tem promovido a alfabetização em *software* nas artes visuais e a alfabetização visual

¹ <https://processing.org/>

² <https://p5js.org/>

na tecnologia. Existem dezenas de milhares de estudantes, artistas, designers, pesquisadores e amadores que usam o Processing para aprendizagem e prototipagem (Processing, 2021, tradução nossa).

Uma aplicação Java, Processing é ao mesmo tempo uma linguagem de programação e um ambiente de desenvolvimento integrado (IDE). O Código 1 demonstra um simples programa em Processing que desenha um quadrado preto em um fundo branco.

Código 1 - Programa em Processing para desenhar um quadrado preto em um fundo branco

```
size(400, 400);  
background(255);  
noStroke();  
fill(0);  
rect(100, 100, 100, 100);
```

O resultado desse programa bastante simples pode ser visto na Figura 1.

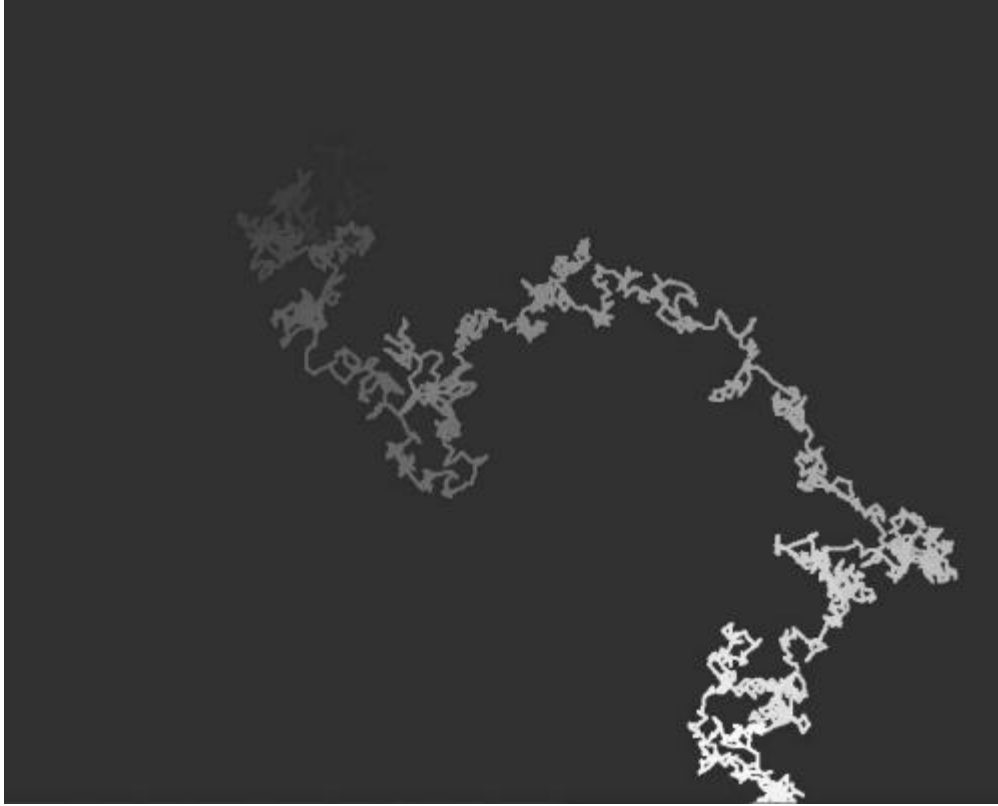
Figura 1 - Quadrado preto em fundo branco



Fonte: Elaborada pela autora

Processing possibilita a escrita de programas baseados em funções, orientados a objetos, ou ainda de forma híbrida. Com uma série de funções predefinidas para desenhar formas geométricas bidimensionais, tridimensionais, esquemas de posicionamento de câmera, de luz, e fórmulas matemáticas, é possível criar os mais variados trabalhos artísticos utilizando Processing. A Figura 2 apresenta um programa mais complexo construído com essa linguagem.

Figura 2 - Linha branca serpenteando um fundo negro



Fonte: Greenberg, 2007

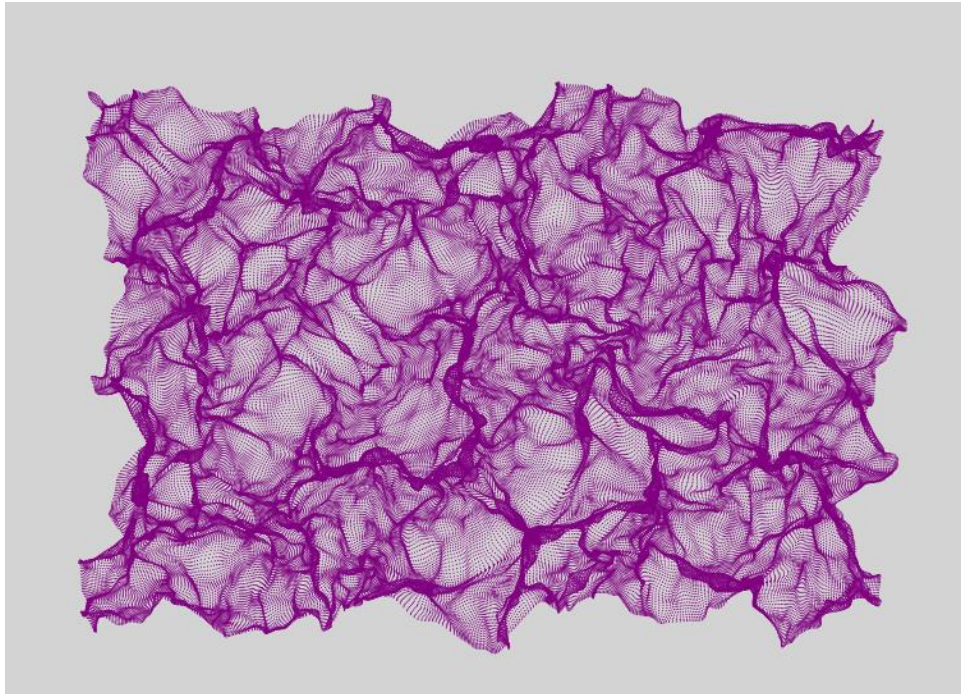
2.1.2 p5.js

A *framework* p5.js é uma ferramenta em javascript para a criação de programas que constroem imagens, animações e interações. A ideia é escrever uma única linha de código e fazer com que um círculo apareça na tela. Adicionando mais linhas, o círculo segue o mouse. Outra linha adicional e o círculo muda de cor quando o mouse é pressionado. O usuário escreve uma linha, depois outra, e assim por diante. O resultado é um programa criado uma peça de cada vez (MCCARTHY, REAS e FRY, 2015).

A biblioteca p5.js consiste em diversas ferramentas que trabalham juntas em diversas combinações. Além disso, existem bibliotecas que estendem o p5.js ainda mais a domínios que incluem reprodução de som, adição de botões, *sliders*, caixas de entrada, captura de câmera, leitura de arquivos, entre diversas outras possibilidades.

A Figura 3 representa uma figura conhecida como *wobble*, ou oscilação, a qual consiste em uma grande quantidade de linhas dispostas lado a lado e seguindo direções semialeatórias.

Figura 3 - Oscilação de linhas roxas sobre o fundo cinza



Fonte: Elaborada pela autora

O código responsável por construir essa figura tem menos de 20 linhas, consiste na chamada de 4 funções que realizam comandos básicos como laços de repetição, expressões numéricas simples e uso de valores semialeatórios, como visto no Código 2.

Código 2 - Programa em p5.js para criar uma oscilação semialeatória

```
function setup() {  
  simple();  
  colorMode(HSB);  
}  
  
function draw() {  
  noStroke();  
  fill('purple');  
  repeat(100, 700, 1, 100, 500, 1, noisy_stripe);  
}  
  
function noisy_stripe(x, y) {
```

```
var x2 = x - 50 + (100 * noise(y / 50, x / 50));  
var y2 = y - 50 + (100 * noise(x / 50, y / 50));  
circle(x2, y2, 1);  
}
```

As funções *setup* e *draw* são os elementos básicos de um programa do p5.js (e também do Processing). Enquanto o *setup* é reservado para os comandos referentes à configuração inicial do programa. Essa função é chamada somente uma vez no início do programa. A função *draw*, por outro lado, roda continuamente de cima para baixo até que o programa seja finalizado.

Com o p5.js, é possível criar uma enorme variedade de programas criativos com o uso de programação. Seu principal foco é a construção de trabalhos imagéticos, mas, como visto anteriormente, pode também usar som, arquivos, entre outros. Entretanto, não possui foco em texto e, portanto, não possui suporte específico para a escrita de histórias.

2.2 NARRATIVAS

Desde os primórdios da humanidade, histórias sempre foram uma forma encontrada pela sociedade para construir comunidades, o que vai desde uma tribo agrupada em volta de uma fogueira até uma comunidade global reunida diante de um televisor (MURRAY, 2003 *apud* DUBIELA e BATTAIOLA, 2007). Histórias fazem parte do ser humano, são intrínsecas no seu modo de viver e interagir em sociedade. Os seres humanos são contadores naturais de histórias.

De acordo com Stephens (1992), citado por Galvão (2005), a narrativa pode ser definida como o conjunto de três componentes: (1) história, abrange os personagens envolvidos em determinados acontecimentos, num espaço e tempo determinados e possibilita uma primeira interpretação do que é contado; (2) discurso, forma específica como qualquer história é apresentada, e (3) significação, uma interpretação de segundo nível que o ouvinte/leitor/espectador obtém a partir do inter-relacionamento da história e do respectivo discurso.

É visível a diferença entre história e narrativa. Enquanto a história trata do fenômeno, o método que a instiga e a descreve se concretiza numa narrativa (CONNELLY e CLANDININ, 1990 *apud* GALVÃO, 2005).

Muitas vezes, as pessoas associam narrativas a palavras, mas elas também configuram gestos, pensamentos, sentimentos, imagens etc. Os gêneros e formas dos textos narrativos parecem ser inumeráveis. Ao comunicar algo sobre um evento da vida, a comunicação geralmente assume a forma da narrativa.

Subcategorias de narrativa incluem mitos, contos populares e contos de fadas, histórias reais ou fictícias e certos textos históricos, jurídicos, religiosos, filosóficos e científicos. Narrativas também podem ser apresentadas na forma de — ou como — poesia, épicos tradicionais e literários, teatro, música, filme, balé e diversificadas formas de arte visual (BROCKMEIER e HARRÉ, 2003).

As tecnologias digitais propiciaram às histórias inovadores meios de transmissão, o que atribuiu às narrativas características singulares tais como a interatividade. Assim, é possível classificar histórias pela sua estrutura geral, a qual pode ser determinada por quais tipos de escolhas estão disponíveis para o leitor, quão abertas ou restringidas são essas escolhas e quais são os efeitos que elas têm tanto na história em desenvolvimento quanto no final. Em conformidade com Beatriz, Martins e Alves (2009, p. 1):

A forma de narrar recebeu outra roupagem ao ser introduzida nos jogos eletrônicos, pois possibilitou a oportunidade de contar histórias com a participação do usuário, que deixa de ser leitor passivo e passa a ser um agente participante da trama.

Os jogos eletrônicos popularizaram uma forma de contar histórias chamada de narrativa interativa, devido a necessidade natural de possibilitar que o jogador interaja e influencie no desenvolvimento da história.

2.2.1 Narrativas interativas

A narrativa interativa é uma forma de entretenimento digital na qual o usuário cria ou influencia o enredo de uma história através de ações, seja assumindo o papel de personagem em um dado universo, interagindo com elementos ou emitindo comandos para um personagem autônomo (RIEDL e BULITKO, 2012).

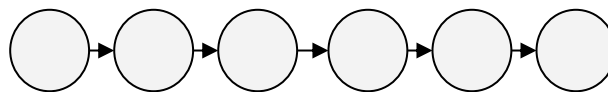
Em uma narrativa linear, a narrativa se desenvolve de maneira única, onde cada acontecimento direciona a outro acontecimento pré-determinado, sem que o leitor crie ou influencie a linha narrativa. Em uma narrativa interativa, no entanto, o leitor possui um número

variável de pontos de escolha, os quais criam ramificações na narrativa de acordo com a decisão tomada pelo leitor.

Embora muito seja falado sobre narrativas interativas no contexto digital, essa modalidade de entretenimento na verdade surgiu no ano de 1979, com uma série de livros chamada *Choose Your Own Adventure* (em português, escolha a sua própria aventura, numa tradução literal). Nesses livros-jogo, o leitor assume o papel do personagem em primeira pessoa e controla o desenvolvimento da história com suas escolhas.

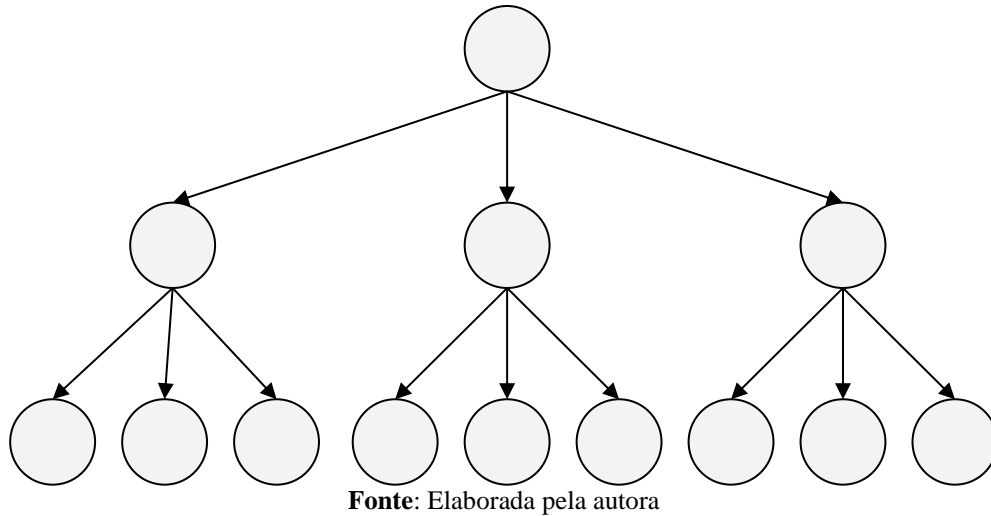
Um jogador possui n opções disponíveis em um ponto de escolha, e cada opção apresenta um desenrolar único da narrativa que possui novos pontos de escolha, ou seja, cada decisão tomada pelo jogador direciona-o para uma ramificação da história que possui novas decisões a serem tomadas, e cada uma destas possui também suas próprias decisões subsequentes. Em uma narrativa linear, na qual cada acontecimento leva a um, e somente um, acontecimento e, portanto, pode ser representada por um grafo linear (Figura 4), o tempo de desenvolvimento é proporcional ao comprimento desse grafo. Uma narrativa interativa, por outro lado, representada por um grafo não-linear (Figura 5), tem seu tempo de desenvolvimento diretamente afetado pelo grau de seus vértices.

Figura 4 - Grafo ilustrando estrutura de uma narrativa linear



Fonte: Elaborada pela autora

Figura 5 - Grafo ilustrando estrutura de uma narrativa interativa



2.3 TRABALHOS RELACIONADOS

Embora seja uma área relativamente pouco explorada, as narrativas interativas estão ganhando considerável espaço no universo acadêmico.

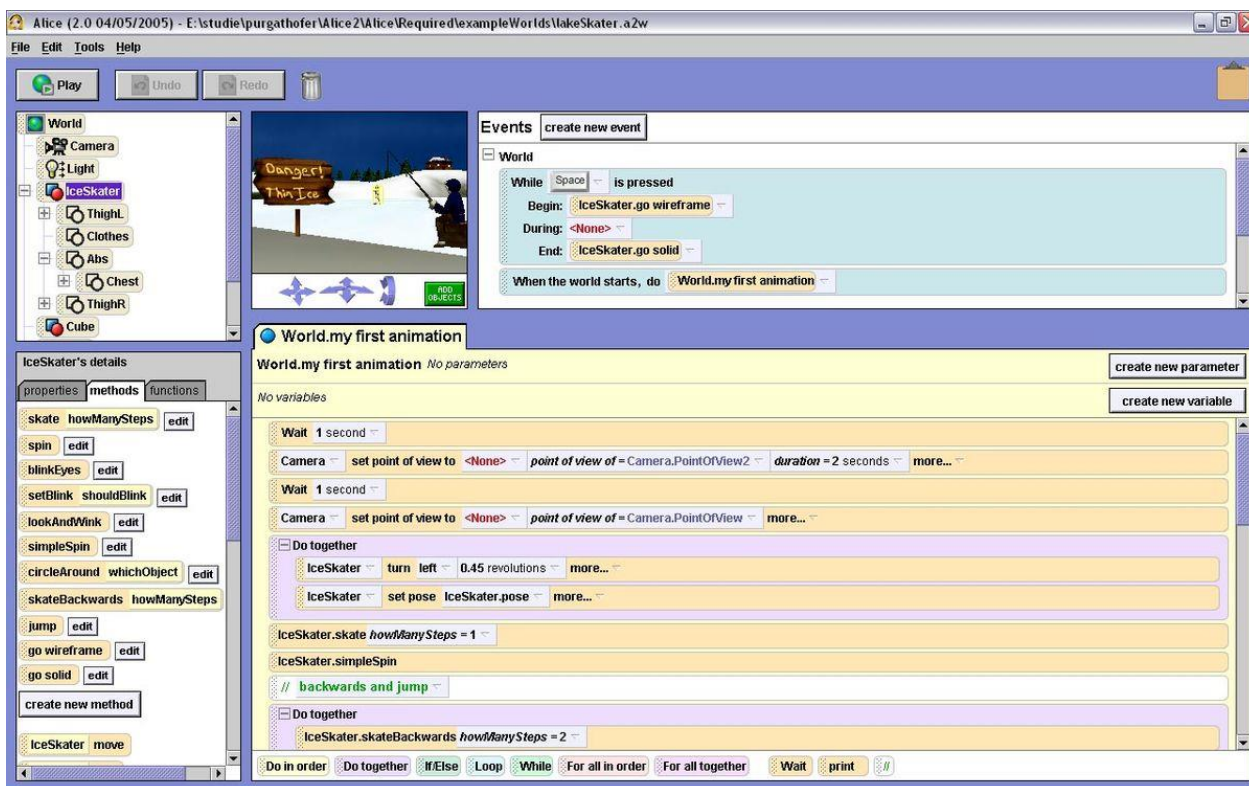
Existem estudos que investigam a utilização dessas narrativas na abordagem de problemas sérios, tais como Time Mage (YIN, RING e BICKMORE, 2012), que tem o propósito de aumentar o empoderamento de pacientes de hospital; Lucidity (GILLIAM et al., 2016), uma narrativa interativa que visa disseminar conhecimento e comunicação sobre violência sexual e temas relacionados à saúde; Migrate (PARROTT, CARPENTIER e NORTHUP, 2017), um jogo em que o jogador pode fazer o papel de um imigrante que cruza ilegalmente a fronteira do México para os Estados Unidos da América; Chimeria: Grayscale (ORTIZ e HARRELL, 2018), que aborda o tema do sexismo no local de trabalho; e Magnet (DA SILVA, JANG e AMARAL, 2020), que retrata relacionamentos abusivos.

Além disso, existem alguns trabalhos que tratam do uso de narrativas interativas para o ensino de programação. Fields *et al.* (2014) propõem uma série de oficinas colaborativas de construção de narrativas interativas utilizando Scratch³, uma linguagem de programação desenvolvida pelo Media Lab do MIT, a qual os autores concluíram ser uma plataforma significativa para usar técnicas de programação sofisticadas em um contexto criativo, baseado em histórias.

³ <https://scratch.mit.edu/>

Kelleher e Pausch (2007) apresentam Alice, uma ferramenta gráfica para a construção de narrativas interativas visuais com programação gráfica. Esse ambiente de desenvolvimento focado no público feminino propõe a criação de animações 3D interativas ou jogos simples utilizando programação em blocos e programação orientada a objetos. Esses autores defendem que contar histórias é uma maneira natural de introduzir os conceitos que fundamentam a programação orientada a objetos.

Figura 6 - Interface gráfica do software Alice 3



Fonte: Wikipedia⁴

Em Cheong *et al.* (2008) é apresentado o PRISM, uma *framework* visual para o desenvolvimento automatizado de narrativas interativas. Seu grande diferencial, em comparativo com os softwares apresentados anteriormente, é justamente essa automatização. PRISM é uma ferramenta autora que utiliza grafos ramificados e inteligência artificial para, a partir de um script fornecido pelo usuário, gerar conteúdo para variados desdobramentos de uma história.

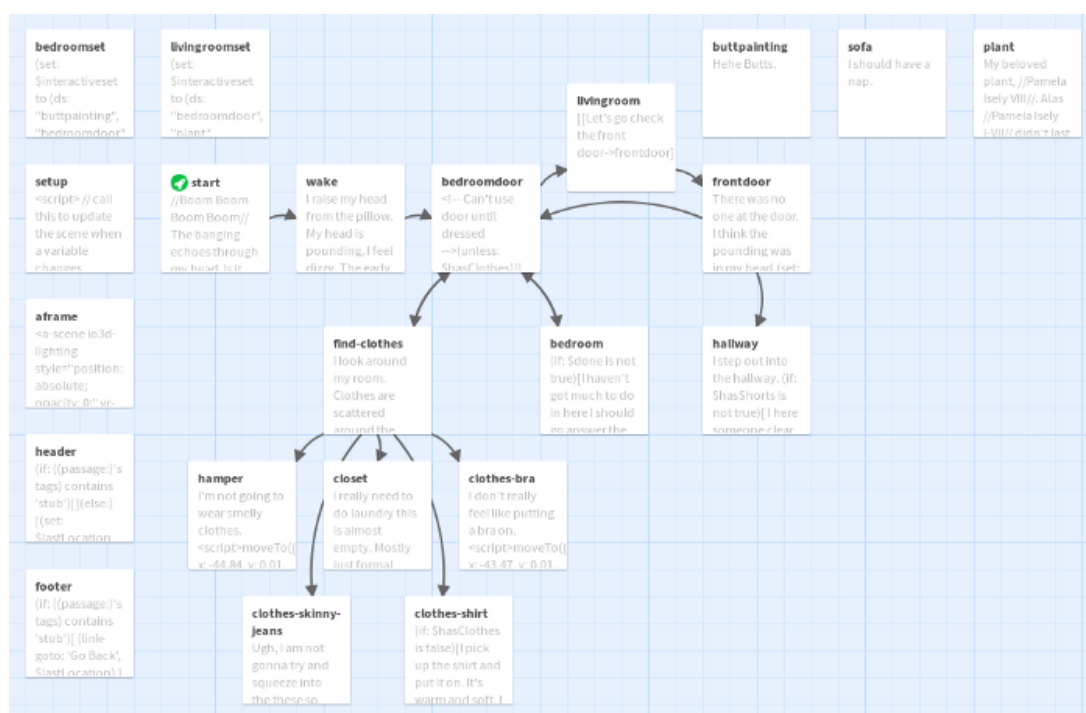
Kim *et al.* (2011) segue a mesma linha do PRISM, ao propor uma ferramenta autora, porém seu diferencial é desenvolver uma linguagem de marcação inspirada em XML. Em seu

⁴ [https://en.wikipedia.org/wiki/Alice_\(software\)](https://en.wikipedia.org/wiki/Alice_(software))

trabalho, é apresentada a SML, sigla de *Storytelling Markup Language* — ou linguagem de marcação de narrativa.

Twine⁵ é uma plataforma que suporta a criação de histórias interativas que possuem narrativas ramificadas e estados variáveis (FRIEDHOFF, 2013). É um *software* visual que propõe simplificar a criação de narrativas interativas, e tenta tornar o desenvolvimento de histórias mais similar o possível do processo real, deixando-o mais análogo ao ato de escrever que ao ato de codificar, como pode ser visto na Figura 7.

Figura 7 - Interface gráfica do Twine



Fonte: Medium⁶

Além disso, Chris Klimas, o criador do Twine, também desenvolveu uma ferramenta chamada Twee, que nada mais é do que uma versão de linha de comando do Twine.

RenPy⁷ é uma ferramenta para a criação de narrativas interativas. Uma história é escrita em um arquivo de texto e a estrutura da história é criada com menus, saltos e rótulos

⁵ <https://twinery.org/>

⁶ <https://medium.com/samsung-internet-dev/how-i-built-a-game-in-a-week-5810b1197686>

⁷ <https://www.renpy.org/>

(LETONSAARI, 2019). O Código 3 abaixo apresenta um trecho de código em RenPy, no qual duas opções são apresentadas ao jogador.

Código 3 - Script em RenPy

```
s "Tá bom, mas o que é uma \"narrativa interativa?\""  
menu:  
    "É um jogo.":  
        jump jogo  
    "É uma história não-linear":  
        jump historia  
label jogo:  
    m "É um tipo de jogo que você pode jogar no seu computador."
```

Letonsaari (2019) também apresenta Inkle, um *software* que combina arte visual e histórias não-lineares, e utiliza a linguagem de script Ink⁸; e ChoiceScript⁹. Ambos são linguagens de *script* textuais de código aberto. O Código 4 demonstra um script em Ink.

Código 4 - Script em Ink

```
=== cena_1 ===  
Você se encontra em frente aos portões do Colégio Thermidor.  
* [Abrir os portões] -> cena_2  
* [Destruir os portões] -> cena_3  
* [Voltar para o portal] -> cena_4  
=== cena_2 ===  
Você abre o portão
```

Além disso, vale a pena ainda mencionar o famoso jogo Episode Interactive¹⁰. É um aplicativo de celular bastante popular que permite que seus usuários consumam e compartilhem suas próprias histórias. Com mais de 100 milhões de downloads no Google Play, Episode possui um catálogo diverso de histórias visuais interativas, podendo ser oficiais, escritas pelo time do jogo, ou escritas pelos próprios jogadores.

⁸ <https://github.com/inkle/ink>

⁹ <https://github.com/dfabulich/choicescript>

¹⁰ <https://www.episodeinteractive.com/>

Para permitir que os usuários escrevam suas próprias histórias, Episode possui sua própria linguagem de *script*, que além de permitir a ramificação da história, também possibilita o uso de animações, imagens, personagens customizáveis, entre outras coisas. No Código 5 é possível ver como pode ser definido um ponto de escolha em uma história.

Código 5 - Trecho de uma cena com duas escolhas para o Episode

```
&MADELEINE starts talk_happy_agree
    MADELEINE
        O príncipe estará lá. Tem certeza de que não vai ao baile?
choice
"Seja sarcástica" {
&RAVENA starts talk_excited
    RAVENA
        Agora eu tenho ainda mais certeza.
} "Seja animada" {
    RAVENA
        Você acabou de dizer as palavras mágicas.
}
```

Conforme o exposto, existem diversas maneiras de desenvolver narrativas interativas. Desde ferramentas autoras, que são capazes de construir novos desdobramentos para a história, até ferramentas gráficas que suportam a criação de personagens 2D ou 3D, a utilização de efeitos sonoros e animações. Todas essas ferramentas possuem em comum o fato de terem sintaxe própria, em geral bastante diferente uma da outra. Comparando Ink e RenPy, Ink possui uma estrutura bastante simbólica, enquanto RenPy, baseado na linguagem de programação python, faz uso de símbolos e palavras-chave.

Portanto, todas as plataformas para escrita de narrativas interativas observadas possuem mesmo propósito geral, mesmo que algumas funcionalidades diferentes, e sintaxes bastantes distintas. Além disso, todas são escritas em inglês e documentadas em inglês. Essa carência de ferramentas em português encorajou a autora a desenvolver uma ferramenta para a construção de narrativas no idioma.

3 NINE.JS

Para a escrita de narrativas interativas, um projeto de iniciação científica da Universidade Estadual do Sudoeste da Bahia construiu uma biblioteca chamada Nine.js, e também foi desenvolvida uma identidade visual para o Nine, como pode ser visto na Figura 8.

Figura 8 - Logomarca do Nine



Fonte: Elaborada pela autora

A biblioteca foi desenvolvida nos moldes das *frameworks* *Processing* e *p5.js*. A linguagem escolhida para sua implementação foi JavaScript, uma vez que, segundo Rigues (2020), em seu artigo para o Olhar Digital, é a linguagem de programação mais popular. O objetivo do Nine.js é implementar simples funções na linguagem que podem ser utilizadas para facilmente escrever histórias não-lineares. Desse modo, o usuário tem a possibilidade de utilizar os próprios recursos do JavaScript em conjunto com os métodos disponibilizados pela biblioteca.

A seguir, o desenvolvimento de narrativas interativas com JavaScript e a biblioteca Nine.js será destrinchado a fim de tornar clara a estrutura dessa *framework* e, finalmente, como podem ser escritas histórias não-lineares.

3.1 ESTRUTURA DE UM PROGRAMA EM NINE.JS

Assim como em *p5.js*, como visto no Código 2, um programa em Nine deve conter pelo menos duas funções principais: *configuracoes()* e *historia()*.

A função *configuracoes()* é chamada quando a execução da narrativa inicia. Como o nome indica, deve ser utilizada para aplicar as configurações da narrativa, tais como o tamanho

da fonte, a cor do texto, o nome do autor, entre outras funcionalidades que serão abordadas mais adiante, além de também ser utilizada a fim de inicializar variáveis.

Após definida a função `configuracoes()`, a função `historia()` faz precisamente o que seu nome sugere, ela é a função na qual deve ser escrita a narrativa. Essa função é chamada imediatamente depois de `configuracoes()`. O Código 6 apresenta a estrutura básica de um programa funcional em `Nine.js`.

Código 6 - Estrutura básica de um programa em `Nine.js`

```
function configuracoes() {  
    // Configurações definidas aqui  
}  
  
function historia() {  
    // História definida aqui  
}
```

Ao ser executado, um programa em `Nine.js` retorna uma página Web com a narrativa interativa, conforme a Figura 9.

Figura 9 - Programa `Nine.js` em execução

Título da História

Nome do Autor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed dapibus turpis quis fringilla interdum. Aliquam at nisl pellentesque, lobortis lacus vitae, pretium ipsum. Morbi vitae purus eget mi facilisis ornare. Aenean vel venenatis magna. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nullam vel augue vel erat consequat ultricies.

Duis quis justo enim. Sed dui ex, pellentesque eu bibendum sit amet, mattis in neque. Phasellus auctor tristique dui, id mollis odio egestas id. Phasellus vestibulum ornare pellentesque. Morbi rutrum vel est vestibulum vestibulum. Fusce at nulla dapibus, mollis mi non, dignissim mi. Phasellus efficitur venenatis urna, vitae rutrum leo posuere sed. Donec cursus feugiat mi. Duis mollis nec justo id porta.

Lorem Ipsum

Aliquam at nisl

Fonte: Elaborada pela autora

3.2 FUNÇÕES DE CONFIGURAÇÕES

A seguir serão apresentadas as funções que podem ser utilizadas dentro do escopo de `configuracoes()`.

3.2.1 Título

A função `titulo()`, como o nome indica, define o título da história, que é exibido conforme pode ser visto na Figura 9. Se um único argumento é fornecido, o valor passado por este é definido como o título da história. Se não houver argumentos, entretanto, a função retorna o título da história, caso já tenha sido definido um, ou *undefined* caso contrário.

Código 7 - Função `titulo()` do `Nine.js`

```
function configuracoes() {  
  titulo("A Supremacia Knight");  
  const retorno = titulo(); // Retorna "A Supremacia Knight"  
}
```

3.2.2 Autor

A função `autor()`, por outro lado, define o nome do autor da história, que é exibido logo após o título, como pode ser visto na Figura 9. Se um único argumento é fornecido, o valor passado por este é definido como o autor da história, e, se não forem passados argumentos, a função retorna o nome do autor definido.

Código 8 - Função `autor()` do `Nine.js`

```
function configuracoes() {  
  autor("Natalia Pinheiro");  
  const retorno = autor(); // Retorna "Natalia Pinheiro"  
}
```

3.2.3 Cor

A função `cor()` define a cor do texto da história. Essa função é tipicamente usada dentro de `configuracoes()`, mas também pode ser usada no escopo de `historia()`. Nesse último caso, ela define a cor do texto para todos os textos abaixo de sua declaração. No Código 9 abaixo, a função `cor()` define que o texto da história será de cor branca.

Código 9 - Função *cor()* do Nine.js

```
function configuracoes () {  
    cor ("#FFFFFF");  
}
```

3.2.4 Cor de fundo

A função *corDeFundo()* define a cor do fundo do texto da história. Funciona da mesma maneira que a função *cor()*, apresentada no item 3.2.3. No Código 10 abaixo, a função *cor()* define que a cor do fundo da história será de cor rosa.

Código 10 - Função *corDeFundo()* do Nine.js

```
function configuracoes () {  
    corDeFundo ("pink");  
}
```

3.2.5 Tamanho da fonte

A função *tamanhoFonte()* define o tamanho da fonte do texto da história. Funciona da mesma maneira que a função *corDeFundo()*, apresentada no item 3.2.4. No Código 11 abaixo, a função *tamanhoFonte()* define o tamanho da fonte do texto como 16 pixels.

Código 11 - Função *tamanhoFonte()* do Nine.js

```
function configuracoes () {  
    tamanhoFonte ("16px");  
}
```

3.3 FUNÇÕES DA HISTÓRIA

A seguir serão apresentadas as funções que podem ser utilizadas dentro do escopo de *historia()*.

3.3.1 Adicionar texto

A função *adicionarTexto()* adiciona um parágrafo de texto na sequência da história. Essa função é usada dentro da função *historia()*, e recebe um argumento: o texto a ser adicionado.

Código 12 - Função *adicionarTexto()* do Nine.js

```
function historia() {  
    adicionarTexto("Era de manhã. Charlotte estava caminhando pela floresta, distraída como de costume.");  
}
```

3.3.2 Adicionar seção

A função *adicionarSecao()* adiciona um título de seção na sequência da história. Essa função é usada dentro da função *historia()*, e recebe um argumento: o título da seção a ser criada.

Código 13 - Função *adicionarSecao()* do Nine.js

```
function historia() {  
    adicionarSecao("Parte I");  
}
```

3.3.3 Adicionar imagem

A função *adicionarImagem()* adiciona uma imagem na sequência da história. Essa função é usada dentro de *historia()* e recebe até três argumentos: o caminho ou url da imagem, largura da imagem e altura da imagem. Os dois últimos são opcionais.

Código 14 - Função *adicionarImagem()* do Nine.js

```
function historia() {  
    adicionarImagem("portal_ilha.png", "200px");  
}
```

3.3.4 Adicionar opções

A função *adicionarOpcoes()* é responsável por criar as ramificações da narrativa. Ela adiciona botões com opções na sequência da história. Essa função é usada dentro da função *historia()*, e recebe um vetor de opções. Para criar opções, deve ser utilizada a função auxiliar *criarOpcao()*.

3.3.4.1 Criar opção

A função `criarOpcao()` cria um botão de opção para a narrativa. Essa função recebe pelo menos dois argumentos: o *label* da opção, que é um texto descritivo de como a opção desenvolveria a história, e uma função, que será executada caso essa opção seja escolhida.

Ainda, a função `criarOpcao()` pode receber um terceiro argumento opcional, as opções, as quais se referem a um objeto com atributos de configurações. As opções suportadas são:

- *inativa*: recebe *false* caso a opção esteja ativa, nesse caso é possível interagir com ela, e *true* caso contrário.
- *cor*: define a cor do *label* do botão.
- *corDeFundo*: define a cor de fundo do botão.

O Código 15 demonstra a utilização das funções `adicionarOpcoes()` e `criarOpcao()`.

Código 15 - Função `adicionarOpcoes()` do `Nine.js`

```
function historia() {
  const _baterNaPorta = criarOpcao("Bater na porta",
  baterNaPorta, {
    cor: "#FFFFFF",
    corDeFundo: "red",
  });
  const _irEmbora = criarOpcao("Ir embora", irEmbora);
  adicionarOpcoes([_baterNaPorta, _irEmbora]);
}

function baterNaPorta() {
  // Desenvolvimento da história caso o leitor decida bater na
  porta
}

function irEmbora() {
  // Desenvolvimento da história caso o leitor decida ir embora
}
```

3.3.5 Entrada

A função `ler()` permite a entrada de dados na narrativa. Ela adiciona um *label*, um *input* e um botão na sequência da história. Essa função é usada dentro de `historia()`, e recebe dois argumentos: um texto, que é colocado como o rótulo explicativo de qual dado deve ser fornecido, e uma função.

Após o leitor inserir um dado e apertar o botão, a função passada como argumento é chamada e o parâmetro dessa função é o dado fornecido pelo usuário.

Código 16 - Função `ler()` do `Nine.js`

```
function historia() {  
  ler("Qual é o seu nome?", continuacao);  
}  
  
function continuacao(nome) {  
  adicionarTexto("O seu nome era " + nome);  
}
```

3.4 PERSONAGENS

Conforme defendido por Kelleher e Pausch (2007), é natural unir a escrita de histórias com programação orientada a objetos. Esse paradigma é suportado pela linguagem JavaScript e pode ser utilizado em conjunto com o `Nine.js` para criar personagens.

Código 17 - Criação de personagem com JavaScript

```
let leslieChristie = {  
  nome: "Leslie Christie",  
  sobrenome: "Dressler",  
  idade: 20,  
  forca: 0.8,  
  raca: {  
    muthano: true,  
    poderes: ["transferencia"],  
  }  
}
```

Seguindo essa mesma linha de raciocínio, é possível criar também cenários, objetos, cenas, entre outros elementos importantes.

3.5 ESTADO

Ao escrever narrativas interativas, pode ser necessário armazenar informações quanto ao estado da história para uso posterior. No Código 18 é apresentado um trecho de narrativa interativa no qual são utilizadas algumas informações guardadas no objeto inventário e do objeto criado no Código 17.

Código 18 - Utilização do estado da história

```
let inventario = {
  faca: true,
  mapa: false,
  chave: false,
  pedra: true,
}

function atravessarOPortal() {
  adicionarTexto("Após sair do carro e se despedir do seu
companheiro de aventura, você se caminha alguns bons minutos em
direção ao castelo. Como você já imaginada, a porta estava
trancada.");

  const _destrancarPorta = criarOpcao(
    "Destrancar porta",
    destrancarPorta, {
      inativa: !inventario.chave, // Inativa se o leitor nao tem
a chave
    });
  const _entrarPorJanela = criarOpcao(
    "Entrar pela janela",
    entrarPorJanela, {
      inativa: !inventario.pedra, // Inativa se o leitor nao tem
uma pedra
    });
}
```

```
});  
const _arrombarPorta = criarOpcao(  
  "Arrombar porta",  
  arrombarPorta, {  
    inativa: leslieChristie.forca < 0.9, // Inativa se o leitor  
    nao eh forte o suficiente  
  });  
const _baterNaPorta = criarOpcao(  
  "Bater na porta",  
  baterNaPorta  
);  
  
adicionarOpcoes([  
  _destrancarPorta,  
  _entrarPorJanela,  
  _arrombarPorta,  
  _baterNaPorta  
]);  
}
```

4 CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi apresentar uma ferramenta que permita a criação de narrativas interativas com código. Para isso, foi realizada uma exploração dos conceitos de programação criativa e narrativa interativa, além de um mapeamento não sistemático das ferramentas existentes que possibilitam a união de ambos.

A biblioteca Nine.js possibilita a escrita de narrativas interativas com a linguagem de programação JavaScript, com funções e documentação escritas inteiramente em português. Por ser baseada em JavaScript, ela não se prende a uma sintaxe específica e construída somente para essa finalidade, e permite que os usuários explorem novas possibilidades.

Com as funções e estruturas apresentadas no capítulo anterior, é possível construir uma grande quantidade de narrativas interativas. Utilizando, ainda, estruturas de repetição, condicionais, constantes, números aleatórios, estruturas de dados, entre outros, é possível construir programas ainda mais robustos.

O código fonte do Nine.js está disponível no GitHub¹¹ e aceita contribuições. Por ter sido escrito em JavaScript, é bastante escalável.

O objetivo dessa biblioteca é simplesmente servir como uma forma de transformar código em narrativas interativas. Para trabalhos futuros, sugere-se a construção de um ambiente de desenvolvimento que possibilite a escrita colaborativa de histórias não-lineares. Existem diversos estudos que aliam a programação criativa com o ensino de programação, uma sugestão é utilizar o Nine.js para esse propósito, uma vez que contar histórias é uma habilidade intrínseca do ser humano.

¹¹ <https://github.com/natalhando/NineJs/blob/main/nine.js>

REFERÊNCIAS

- ACKERMANN, Judith; EGGER, Benjamin; SCHARLACH, Rebecca. Programming the postdigital: Curation of appropriation processes in (collaborative) creative coding spaces. **Postdigital Science and Education**, v. 2, n. 2, p. 416-441, 2020.
- BEATRIZ, Isa; MARTINS, Jodeilson; ALVES, Lynn. A crescente presença da narrativa nos jogos eletrônicos. In: **VIII Brazilian Symposium on Games and Digital Entertainment**. 2009.
- BROCKMEIER, Jens; HARRÉ, Rom. Narrativa: problemas e promessas de um paradigma alternativo. **Psicologia: reflexão e crítica**, v. 16, n. 3, p. 525-535, 2003.
- CHEONG, Yun-Gyung et al. Prism: A framework for authoring interactive narratives. In: **Joint International Conference on Interactive Digital Storytelling**. Springer, Berlin, Heidelberg, 2008. p. 297-308.
- CORMEN, Thomas; LEISERSON, Charles; RIVEST, Ronald. **Algoritmos**. Elsevier Brasil, 2017.
- DA SILVA, Natalia Pinheiro; JANG, Jenifer de Jesus; AMARAL, Vinicius Oliveira. Magnet: exposing abusive relationships with interactive narrative. In: **Anais do I Workshop sobre as Implicações da Computação na Sociedade**. SBC, 2020. p. 145-152.
- DUARTE, Fábio. Arquitetura e tecnologias de informação: da revolução industrial à revolução digital. **Annablume**, São Paulo, 1999.
- DUBIELA, Rafael Pereira; BATTAIOLA, André Luiz. A importância das Narrativas em jogos de Computador. In: **Proceeding of the Brazilian Symposium on Games and Digital Entertainment (SBGames 2007)**. 2007.
- DUFVA, Tomi Slotte. Creative Coding as Compost(ing). In: **Post-Digital, Post-Internet Art and Education**. Palgrave Macmillan, Cham, 2021. p. 269-283.
- FIELDS, Deborah A. et al. Interactive storytelling for promoting creative expression in media and coding in youth online collaboratives in Scratch. **Proceedings of constructionism**, p. 19-23, 2014.
- FRIEDHOFF, Jane. Untangling Twine: A Platform Study. In: **DiGRA conference**. 2013.
- GALVÃO, Cecília. Narrativas em educação. **Ciência & Educação (Bauru)**, v. 11, n. 2, p. 327-345, 2005.
- GILLIAM, Melissa et al. “Because if we don’t talk about it, how are we going to prevent it?”: Lucidity, a narrative-based digital game about sexual violence. **Sex Education**, v. 16, n. 4, p. 391-404, 2016.

GONZÁLEZ, Alberto Pacheco; PRECIADO, Edgar Trujillo; TERRAZAS, José Alberto Domínguez. PROGRAMACIÓN CREATIVA. 2015.

GREENBERG, Ira. **Processing: creative coding and computational art**. Apress, 2007.

KELLEHER, Caitlin; PAUSCH, Randy. Using storytelling to motivate programming. **Communications of the ACM**, v. 50, n. 7, p. 58-64, 2007.

KIM, SeokKyoo. *et al.* Programming the story: Interactive storytelling system. **Informatica**, v. 35, n. 2, 2011.

LETONSAARI, Mika. Nonlinear Storytelling Method and Tools for Low-Threshold Game Development. In: **Seminar. net**. 2019.

MCCARTHY, Lauren; REAS, Casey; FRY, Ben. Getting started with P5.js: Making interactive graphics in JavaScript and processing. **Maker Media**, Inc., 2015.

MCCORMACK, Jon *et al.* Ten questions concerning generative computer art. **Leonardo**, v. 47, n. 2, p. 135-141, 2014.

ORTIZ, Pablo; HARRELL, D. Fox. Enabling critical self-reflection through roleplay with chimeria: Grayscale. In: **Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play**. 2018. p. 353-364.

PARROTT, Scott; CARPENTIER, Francesca R. Dillman; NORTHUP, C. Temple. A test of interactive narrative as a tool against prejudice. **Howard Journal of Communications**, v. 28, n. 4, p. 374-389, 2017.

Processing, 2021. Disponível em: <https://processing.org/>. Acesso em: 02 de nov. 2021.

RIEDL, Mark; BULITKO, Vadim. Interactive narrative: A novel application of artificial intelligence for computer games. In: **Twenty-Sixth AAAI Conference on Artificial Intelligence**. 2012.

RIGUES, Rafael. JavaScript se consolida como a linguagem de programação mais popular. **Olhar Digital**, 2020. Disponível em: <https://olhardigital.com.br/2020/10/22/noticias/javascript-se-consolida-como-a-linguagem-de-programacao-mais-popular/>>. Acesso em: 07 de nov. de 2021.

SOBREIRA, Elaine Silva Rocha; DOS SANTOS, Veronica Gomes; SOBREIRA, Edson Sidnei. Computação criativa com Scratch, Mixly e Arduino: Prototipando com HackEduca Conecta. **Sociedade Brasileira de Computação**, 2020.

YIN, Langxuan; RING, Lazlo; BICKMORE, Timothy. Using an interactive visual novel to promote patient empowerment through engagement. In: **Proceedings of the International Conference on the foundations of digital Games**. 2012. p. 41-48.