

CONSIDERAÇÕES ACERCA DO CÓDIGO FONTE NA POESIA DIGITAL

Otávio Guimarães TAVARES*

- **RESUMO:** Este artigo tem como objetivo mapear e analisar alguns aspectos da ciência da computação, quais sejam, no como afetam a criação, fruição estética e modo de operar da poesia digital. Especificamente, trata-se de olhar a linguagem de programação como parte da criação e constituição da obra literária digital, como aquilo que está por trás do que é mostrado ao leitor, como uma restrição e potencialidade da criação, e enquanto um fator signficante em uma construção que tem sua própria materialidade como doadora de sentido. Por fim, trata-se de tentar compreender até que ponto o conhecimento desses modos e meios são necessários tanto para criação quanto para fruição das obras literárias digitais.
- **PALAVRAS-CHAVE:** Literatura digital. Tecnologia. Ciência da computação. Linguagem de programação. Criação. Fruição estética.

Código-fonte

Quando lemos um poema digital como o *Amor de Clarice* (TORRES, 2005) ou o *Palavrador* (MARINHO et al., 2006) olhamos a face externa da obra literária digital, ou melhor, olhamos como ela se apresenta ao leitor da mesma forma que se observa um quadro. Porém, há outro modo de olhar um objeto literário digital – que obviamente não deve ser confundido com uma apreciação estética – que deve ser empreendido num trabalho de análise. Isso seria olhar a obra quanto a sua construção, não apenas suas operações tecnológicas, mas também o que está por trás destas operações, ou seja, dos modos de operação que são do domínio da ciência da computação – algoritmos, linguagens de programação, sistemas operacionais – esta vertente da ciência que une o conhecimento científico ao funcionamento tecnológico da computação, entre outros fatores que estão aquém da literatura digital. Tal procedimento de análise implica pensar a obra de arte digital um pouco

* Doutorando em Teoria Literária na UFSC – Universidade Federal de Santa Catarina. Centro de Comunicação e Expressão. Pós-Graduação em Literatura. Membro do Núcleo de Pesquisas em Informática, Literatura e Linguística – NUPILL. Florianópolis – Santa Catarina – Brasil. CEP: 88040-970 – nonada1@gmail.com.

Artigo recebido em 14 de outubro de 2010 e aprovado em novembro de 2010.

mais próxima do modo como se pensava a arte renascentista, cuja divisão entre artesão e artista, ciência e arte, não era algo presente no pensamento da época. É ainda, pensar como em uma época em que um artista se ocupava de questões tanto científicas quanto tecnológicas. Logo, vou ao código fonte, esse outro local da escrita, esse outro lado de uma visualidade que se mostra na tela, essa escrita de potencialidades que constrói a possibilidade de uma obra, que restringe para se abrir e criar uma obra. Também abordarei as relações do criador e do leitor com a técnica digital, como eles podem tocar aquele espaço que inicialmente interpõe tantas camadas entre eles e a obra.

Necessário?

Antes mesmo de empreender uma tarefa que remonte a um olhar sobre a técnica de composição de uma obra digital, devemos pôr duas questões que tendem a pairar sobre qualquer tentativa de observar o código fonte num contexto de estudos artísticos. A primeira: porque tentar olhar o código fonte se este está por detrás da obra de tal forma que ele não é visto ou tocado pelo leitor? Segundo, não será o código um domínio fora da composição explícita da obra, sendo que até vários autores digitais não programam a suas obras, mas recorrem a um programador para a tarefa (evidenciando um afastamento do autor com relação à técnica de composição ou seria a complexidade que distancia o autor de um conhecimento de uma técnica tão específica e complexa)?

EFE – um breve exemplo

Para começar a responder à primeira questão vejamos um exemplo: a releitura criada por Rodrigo Melo e Pedro Reis (2009) intitulada *Poemas em efe* de uma das partes do *Brin cadeira*¹ de Salette Tavares. O hipotexto consiste de uma série de palavras com a letra F escritas em pequenas estrofes, com diferentes fontes, de vários ângulos na folha. Essas escolhas da autora formam versos, estrofes e relações textuais dentro do espaço da folha.

¹ Poema que foi editado no *Cadernos de Poesia Experimental 1* em 1964. Pode-se acessar a versão digital do periódico, como também a de outras publicações do PO-EX (POESIA EXPERIMENTAL PORTUGUESA, 2009).

Na versão impressa, existe a liberdade do leitor de vagar e escolher percursos dentro da página, de se mexer ou mover o papel fisicamente para conseguir ler, de colocar seu corpo de alguma forma ativa na leitura, mexer – interagir com – a obra material ou a si mesmo para o ato de leitura, e não ficar apenas observando palavras dançando na tela, o que eventualmente tende a se tornar um ruído amorfo e entediante. O problema que vejo aqui não é a leitura no computador, mas o ato de passar de um texto que levava o leitor a pensar o ato de leitura, forçando-o a sair da condição costumeira, para um que não tem força em propor essa revisão de hábito. Enquanto o impresso dava margem a uma interação física, o digital é apenas uma série de palavras flutuando aleatoriamente.

Constatamos que a releitura não deu conta de criar algo com a obra, mas funciona apenas como uma transferência do meio impresso para o digital sem levar em consideração certos fatores do hipotexto. O detalhe que está mais aprofundado na breve análise desse poema digital é, mais do que a simplicidade do código utilizado na programação dessa releitura do poema de Salette (e que poderia passar despercebido), o fato de que o código usado é um *template* padrão que está disponível em vários locais da rede e pode ser aplicado com um rápido Ctrl+c Ctrl+v para qualquer léxico, adicionando a ele quaisquer palavras que agradem o programador, resultando num *flash* parecidíssimo com a “releitura” aqui vista.

Isso nos leva inevitavelmente à pergunta: será que temos aqui realmente um esforço de releitura? Temos aqui um ato de criação artística? Um esforço criativo? Ou simplesmente um copiar e colar que acaba destituindo o poema original de elementos que eram de extremo valor e importância?

Tal questionamento só pode ser feito através de um conhecimento sobre a programação². Esse exemplo permite evidenciar a importância de conhecer minimamente algo sobre a mesma e os principais tipos de mídia utilizados para as criações digitais. Desse modo, é possível fazer uma leitura a fundo do objeto em questão, já que, o meio digital tem como uma de suas características trazer o suporte como parte integrante da construção de sentido (o conhecimento da programação muitas vezes é deixado de lado pelos teóricos e críticos do meio digital, talvez por se tratar de um conhecimento técnico e, como tal, visto muitas vezes como inferior por nós da área das Humanidades).

² Observem que aqui digo “sobre” e não “da”, pois, nesse caso, alguém pode chegar a descobrir sobre a programação dessa releitura sem ter que abri-lo ou sem possuir conhecimento específico de programação, basta procurar uns poucos exemplos de movimento aleatório em algum motor de busca na internete, para encontrar outros arquivos *flash* do mesmo tipo que a releitura. Eu mesmo o fiz em 2007 para montar a capa da revista *Texto Digital*: <<http://www.textodigital.ufsc.br/num05/num05.html>>.

Mostrar-se

O código não se mostra imediatamente, nem sempre ele está acessível por meios “lícitos”. E não é necessário conhecê-lo para fruir a obra, porém, ele pode ser trazido à vista. Eu posso fazê-lo aparecer, manipulá-lo, e logo manipular a obra. Contudo, apesar de falar aqui de código fonte, não temos uma homogeneidade absoluta entre eles, os códigos. Existem milhares de linguagens de programação, desde coisas como o *BASIC*, até linguagens mais complexas como a *Python*, cada uma com suas características e peculiaridades, fins e utilidades.

Na maioria dos casos apresentados nesse artigo, os poemas digitais são construídos usando a linguagem *actionscrip*, utilizando o programa *Flash*, que tem como uma de suas características a possibilidade de integrar vários tipos de mídias de uma forma leve e simples e de ser amplamente distribuído e usado numa grande quantidade de máquinas e sítios. Outras linguagens também utilizadas são, por exemplo, o *html*, *lingo*, entre outros. Há também *toolkits* que são utilizados para criação de obras digitais interativas, como o *Vvvv* e o *Arduino*, com alguns até mesmo desenvolvidos explicitamente com esse propósito, como é o caso do *Pure Data* e o *openFramework* (todos de código livre). O mais importante é saber que, com algumas exceções, os códigos são geralmente fechados para acesso de quem está fruindo a obra (o fruidor não tem como abrir o código naquele momento). O autor cria sua obra num programa que possibilita a escrita com aquela linguagem e quando termina ele fecha o arquivo (que será disponibilizado na internete ou em qualquer outro meio acessível)³.

Logo, eu, enquanto leitor, não posso ver como um ou outro poema foi construído, não tenho acesso a essa face da obra⁴, ela resta subscrita nas entrelinhas do visual, ela apenas permeia aquilo que se mostra, e ao mesmo tempo subjaz por trás de toda materialidade ali exposta. A minha possibilidade de ver aquelas linhas que estão por detrás da materialidade exposta irá depender do meu entendimento do código, não um que seja “livresco-enciclopédico” específico, mas um entendimento enquanto potencialidade de funcionamento, como possibilidade de ação visível, um conhecimento do código em funcionamento.

³ Por exemplo, quando o autor trabalha no programa *Flash*, no momento em que está trabalhando o arquivo é um *.fla* e, assim que ele encerra seu trabalho, ele o exporta para o formato *.swf* (devidamente lacrado).

⁴ Claro que existem programas que facilmente quebram a trava do objeto em *flash*, permitindo acesso ao seu código. No entanto, isso não é o “normal” do código, é uma violação de um laque, o que muitas vezes é visto como uma quebra de direito do autor do arquivo, sendo esses programas de quebra distribuídos com o propósito de permitir que o usuário abra a trava de um de seus próprios arquivos.

Programa dentro de programa dentro de...

Após termos tratado de um código fonte, uma camada da obra digital, é importante estarmos atentos para o fato de que essa relação – código e obra – está contida em outra, porquanto a obra opera num OS – sistema operacional – (como Windows, linux, Mac, etc.). E este OS também foi programado, utilizando outra linguagem e, como a obra digital, ele também tem sua face visível e outra oculta⁵. A obra digital é programada usando certo código fonte (linguagem específica) num programa, rodando depois como um aplicativo. Tudo isso num sistema operacional que também foi programado e possui um código fonte. A programação de uma obra está submetida a fatores operacionais de um programa que está submetido a fatores mais baixos pertencentes ao OS. Na maioria dos poemas as regras do OS permanecem inalteradas pelo autor e o poema continua sendo uma operação dentro de um mundo maior, ou como nos diz Philippe Bootz (2005): *“At a technical level, author and reader are only users of the computer. Notably, the author does not manage in his engagement the totality of the rules that are used by the computer while running”*⁶.

Tanto a programação do programa quanto as regras gerais do OS impõem certas limitações e delimitam fatores no modo de operar da obra (e ao mesmo tempo permitem que ela opere e exista). A obra não subverte o OS⁷; algumas podem operar em tela cheia, como o *Amor de Clarice* (TORRES, 2005), o que não significa anular os outros aparatos que existem no computador, tais como dicionários, calculadoras e os navegadores de internet. O que isso quer dizer é que o computador não opera apenas a obra: esta não toma a totalidade da máquina (me restrinjo aqui ao caso de poemas que aparecem num computador doméstico⁸), pois opera simultaneamente com vários outros objetos. A obra é um código específico dentro de um mundo de regras muito maior.

O que nos interessa é o nível de programas que operam dentro um do outro. Temos um computador que opera um OS, que tem um programa, que permite criar

⁵ O “avanço” de um sistema operacional Macintosh ou Windows com relação ao DOS era justamente o layout visual que dispensava o usuário da necessidade de conhecer os códigos operacionais do DOS, tornando a navegação algo mais fácil para o usuário. E com relação à face oculta é bom ter em mente que os códigos de sistemas operacionais são muito mais difíceis de abrir ou de serem alterados, podendo-se aqui realmente falar de oculto.

⁶ “Num nível técnico, autor e leitor são apenas usuários do computador. Notavelmente, o autor não comanda em seu emprego a totalidade das regras utilizadas pelo computador enquanto este opera” (tradução nossa).

⁷ Não tenho conhecimento de alguma obra que o faça, porém não é difícil de imaginar uma que opere tal feito.

⁸ Pois existem obras digitais que são criadas para operar em instalações ou exposições e que são a única coisa a operar na máquina. Máquina estas que são preparadas justamente para aquela criação. Por exemplo: a versão de livro físico do *Palavrador*, em que o suporte foi até mesmo desenvolvido para a interação daquela obra, como pode ser vista nesse link <http://1maginari0.blogspot.com/2009/03/palavrador.html>.

um pequeno aplicativo que será o poema digital, que é uma obra interativa, que permite a interação do leitor e que com sua leitura chega a um texto. Pensemos assim: Computador (o corpo físico recebendo energia elétrica para operar) => OS (tem nos seus códigos a possibilidade de um programa) => programa (tem no seu código a possibilidade de criar pequenos aplicativos como *actionsript*) => poema digital/obra (poema potencial que tem margem para interação do leitor) => texto (um caminho efetivado pelo leitor).

De um computador com um OS abrimos as possibilidades de criar outras coisas dentro de suas regras. Temos sistemas de regras dentro de outros sistemas de regras, até finalmente encontrarmos o leitor que se propõe a interagir com o sistema de regras chamado obra para sua leitura.

Código é tudo?

Mas então diríamos que o código é a obra? Afinal, ele é a base de toda operação da obra e tudo que nela acontece está escrito em suas linhas. Logo, ele é a obra? Ele é o objeto em questão? Não, o código não é a obra em si, da mesma forma que a tinta de *Les Baigneuses* de Paul Cézanne não é a obra, mas sim parte da obra, do processo que podemos chamar a obra de arte. É uma face da obra, um estágio de sua criação que perdura ao longo do tempo. A obra é um processo do qual a programação faz parte. Não que ela seja apenas processo, ao modo das vanguardas do século XX que colocavam todo o valor da obra de arte no seu caráter processual, ignorando os resultados e renegando-os como algo dispensável ou sem importância para o ato criativo. Não. Quando digo que a obra é um processo, me refiro à totalidade de um processo que englobe tanto o ato de criação quanto o resultado final.

Philippe Bootz (2005) parece chamar a atenção rapidamente para a obra enquanto processo, da programação enquanto um ato criativo que gera um código que irá ser efetivado pela máquina a ser exposta como uma obra e ao mesmo tempo o ato de programar enquanto um de criação artística: *“we consider programming (and not the program) the material of this art, that the multimedia event that appears on screen is only a transient observable state (a “transitoire observable” in French) that occurs while running”*.

Olhemos a questão da programação como uma obra potencial, lembrando aqui justamente as obras impressas do OuLiPo francês que se faziam valer desse mesmo aspecto programático e processual ao propor regras ou *contraintes* para a criação

⁹ “nós consideramos o programar (e não o programa) a matéria dessa arte, do qual o evento multimídia que aparece em tela é apenas o estado transitório observável (“transitoire observable” em francês) que ocorre enquanto operando” (tradução nossa).

literária. Entretanto, a parte potencial está na sua materialidade e será efetivado pelo computador. É como se o programa fosse o plano de obra, o que poderá vir a ser, o conjunto de regras do que será possível, uma vez que ela seja colocada para operar. Assim, realmente podemos conceber a obra enquanto processo, pois ela só existe em movimento, sendo um operar, um trabalhar, um ato; e será nesse operar que o leitor terá contato com ela.

Porém, na verdade, não nos convém pensar o programar como a matéria da arte digital, visto que se trata de uma etapa de uma arte processual; não podemos, contudo, esquecer o resultado final, nem nenhum outro momento de seu processo. Talvez seja de grande ajuda pensarmos o ato de programação e o modo de efetivação da obra digital ao lado das artes performáticas, como a recitação de um poema (que teve por muito tempo o propósito de ser recitado) e ainda a encenação de uma peça teatral. Os versos escritos e memorizados pelos atores não são a totalidade da peça, ela é o conjunto destes no ato de ser encenada, ao se tornar ação e ser efetivada; em outras palavras, não se pode ignorar alguma parte do processo. Deve-se olhá-lo como um todo. Hans-Georg Gadamer (1999) caracteriza a obra de arte como pura mediação, em que não podemos entendê-la como obra se não for no mundo, no seu processo de existir. Um poema ao ser declamado está de alguma forma unido ao ato de confecção primária, em todos os momentos de sua existência. A obra é mais do que um texto ou um objeto, ela é todo um movimento que aponta para um ou mais sentidos, para sua existência como experiência.

Programação de possibilidades

Da mesma forma que uma peça teatral shakespeariana não especifica todas as ações e suas configurações no palco, cada expressão perfeita que um personagem terá que portar em sua face, também o código não necessariamente diz com precisão o que acontecerá. Existem códigos fixos cuja programação diz exatamente o que irá ocorrer com cada variável possível, porém, estes são mais limitados, estão fixados numa série de elementos e não em possibilidades. Em outros, não se programa algo, se programam possibilidades de algo. No *Amor de Clarice* (TORRES, 2005) não temos programadas todas as posições em que cada palavra poderá ser movimentada pelo leitor. Não temos programado explicitamente como cada som irá se sobrepor um ao outro. Temos a possibilidade de movimentar cada elemento (palavra) dentro do eixo X e Y, temos a possibilidade de fazer as palavras soarem. A efetivação total da obra não está toda escrita no código. Nesse sentido, podemos falar do código como uma série de regras que possibilita a existência de um mundo no qual o leitor irá adentrar e interagir dentro das possibilidades programadas.

A programação de possibilidades fica ainda mais clara quando falamos de algoritmos, de uma regra que será permutada com outros elementos possíveis e com outras regras, permitindo uma gama de resultados variados. Quando olhamos uma obra como o *Palavrador* em que percebemos efetivamente a construção de um ambiente digital, podemos notar melhor esse tipo de programação em aberto. Nessa criação não se programam todas as possibilidades de ângulos e visões que o seu cubo poderá ter no mundo, programa-se o mundo, as possibilidades de espaço habitáveis pelo seu personagem/cubo, programa-se um mundo em potencial, programa-se o que ao usuário será permitido fazer. Existem programas que criam possibilidades dentro de possibilidades, e assim por diante, entrando num grau de abstração enorme, a tal ponto que nós não podemos efetivamente ver tudo o que será possível. Como também existem algoritmos genéticos, que seriam sistemas programados para estar em constante mudança (ou melhor, permutação), para se alterar e ir proliferando essas mudanças com o intuito de ser uma mera réplica de um processo vivo.

O que temos com tudo isso é de certa forma um processo de criação que, como a programação literária, propõe elementos e regras de operação entre estes elementos a serem efetivados e utilizados pelo computador e depois pelo leitor; programando-se um campo de possibilidades, deixando em aberto. Mas temos dois estágios: o primeiro, em que o código fonte será efetivado pela máquina (colocado para operar ou tocar); o segundo, que será a interação do leitor com relação ao que é exposto pela máquina. Logo, podemos entender a obra visual que abrimos no computador como a efetivação de um cruzamento de possibilidades em computação pela máquina, e nossa interação como movimentos dentro do possível, nossa interação como efêmera efetivação de um caminho possível dentre os possíveis trazidos pelo código. Ou como nos diz Bootz (2005):

The program written by the author does not totally manage the physical process of running. That is, the algorithmic level of the program is not completely responsible for the functioning of the transient observable. We can say that the author is the author of the program and data, but only the co-author of the physical process that appears to the reader while the machine is running. Using more traditional literary language, we can say that the author's program contains a large level of the "non-said". But this non-said does not play the same role as the non-said of the classical printed text: this non-said will be interpreted by the machine and not by the reader to produce the observable sign. Thus it is necessary to consider that program (algorithmic level) and physical process while running are two different and complementary parts of the work. Adaptive generation uses this unsaid as a constraint for programming.¹⁰

¹⁰ “O programa escrito pelo autor não coordena completamente o processo físico de operar. Isto é, o nível algorítmico do programa não é inteiramente responsável pelo

No entanto, não devemos pensar que o autor seja um co-autor, como quer Bootz, sendo o computador o outro autor. Tal afirmação corre sempre o risco de dar ao computador um estatuto de autor, de recair na ideia, proveniente da ficção científica, de um computador inteligente e consciente, o que está obviamente muito longe de ser o caso atual. A co-autoria, da qual fala Bootz (2005), seria concernente ao fato de o computador calcular e preencher certas lacunas, todavia, tal feito se encontra já no ato de criar do autor. O autor digital pressupõe esse ato computacional, seu movimento criativo já se apropriou do que irá ser feito pelo computador. As ações do OS não são impensadas dentro da obra, mas ao contrário, se tornam tão naturais quanto o efeito de acústica para um músico tocando numa capela. Não há co-autoria por parte da capela ou do computador, mas um elemento que vem ao encontro com o ato criativo do autor, mais um elemento do mundo que seu ato criativo levará em conta sem que seja necessário estar conscientemente pensando todos os aspectos daquela interação ao criar.

Girar das restrições

Com o que já vimos a respeito da programação e do código fonte, podemos falar de duas restrições ao ato criativo do autor: a da criação (como as *constraints* utilizadas pelo grupo OuLiPo) e a restrição técnica (a linguagem de programação utilizada, o programa, a tinta, ou qualquer outra restrição física do meio).

Claro que as duas estão imbricadas. Não tenho como criar certo jogo de leitura sem levar em consideração os meios físicos para conseguir arquitetar aquilo que pretendo. Não há como ignorar a programação no ato de criar no meio digital da mesma forma que não se pode ignorar as vigas ao erguer uma catedral gótica. Sempre haverá elementos a serem moldados e alterados pelo ato criativo, há sempre uma ferramenta, há sempre uma interação, um moldar o mundo. Não há criação sem restrição da técnica e dos objetos e elementos utilizados (a não ser uma criação

funcionamento do transitório observável. Podemos dizer que o autor é o autor do programa e dos dados, mas somente o co-autor do processo físico que aparece ao leitor enquanto a máquina está operando. Usando uma linguagem mais tradicionalmente literária, podemos dizer que o programa do autor contém uma grande quantidade de “não-ditos”. Mas esses não-ditos não têm a mesma função que os clássicos não-ditos do texto impresso: esses não-ditos serão interpretados pela máquina, e não pelo leitor, para produzir o signo observável. Logo, é necessário considerar que o programa (nível algorítmico) e processo físico enquanto operam são duas partes diferentes e complementares da obra. A geração adaptável usa esses não-ditos como uma restrição para a programação”. (tradução nossa).

ex-nihil, se é que tal coisa existe). Todo ato criativo irá ao encontro do mundo. É justamente essa interação com o mundo a minha volta que me permite criar. Criar é estar imerso no mundo.

O autor digital deve se apropriar da ferramenta digital, pois as restrições que a ferramenta impõe ao objeto final também aparecem para o criador como limites para o ato criativo. Pintar um quadro utilizando tinta a óleo ou giz de carvão representa diferentes restrições ao ato de pintar – ou seja, o que é possível fazer com aquele objeto – e às possibilidades da obra final; restrições essas que irão transparecer ao observador do quadro de uma forma ou de outra.

No meio digital podemos falar desse tipo de restrição como sendo a linguagem de programação e o programa utilizado para criar uma obra. Normalmente não se cria uma linguagem de código só para criar um poema, o que é análogo à linguagem verbal/escrita, apesar de que, tanto em nossa linguagem quanto na linguagem de programação, isso é possível (principalmente se pensarmos não em criar algo totalmente novo, mas em alterar profundamente aquilo que já existe), especialmente no digital em que a linguagem de programação não transparece completamente (são mais seus efeitos ou processamentos). Isso se torna ainda mais possível se pensarmos que a linguagem de programação é similar a uma ferramenta, e em qualquer arte é possível encontrar autores que alteram ou moldam sua ferramenta a seu gosto para atingir um certo objetivo, para experimentar ou simplesmente para tornar aquela ferramenta mais sua. Lembremos de Leonardo da Vinci preparando tinta a óleo, para ter outro efeito na tela, em vez do processo mais comum na época com têmpera feita com ovos, ou um guitarrista alterando os *pick-ups* e a grossura das cordas ou até mesmo os pedaços inteiros de sua guitarra para atingir outro som, ou ainda o trompete (à primeira vez, acidentalmente) torto a quarenta e cinco graus de Dizzy Gillespie. E se quisermos pensar na linguagem, pensemos no *Livre* de Mallarmé e sua estranha forma de construção que parece querer reorganizar o ato de escrita.

Assim, todo o aparato escolhido para criar no meio digital acarretará alguma restrição ao processo criativo. Mais especificamente, no meio digital são as linguagens de programação e os respectivos programas que são utilizados para compor nesses códigos. Podemos inclusive pensar em questões estilísticas da obra digital com base na escolha do aparato de produção, vislumbrar um estilo por meio das restrições de programa. O que é possível em *flash* difere um pouco do que é possível em *shockwave director* e é ainda totalmente diferente do que é possível ou intencionado com um *VRML* (que é feito para criar objetos em 3D)!

Cada programa apresenta seu propósito inicial, suas funções e suas possibilidades. O autor terá que escolher e jogar com estes programas e, em outros momentos, alterá-los, trazer para eles elementos estranhos como, por exemplo, vídeos criados com outros programas, ou sons construídos em editores

de som. Terá que tentar de algum modo alterar o funcionamento deles ou explorá-lo para atingir seus objetivos de criação e, ao mesmo tempo, deixar que certas possibilidades do programa influenciem, mas não contaminem involuntariamente seu ato criativo.

Técnica nova

Temos no meio digital um distanciamento aparente entre o ato de programar – criar – e o resultado final de nosso ato. Também temos, aparentemente, um deslocamento considerável entre o que foi efetivamente feito e construído e o que se mostra ao leitor no final do caminho. Este talvez seja o fato mais marcante da criação em meio digital, uma mediação entre o eu que cria e o objeto que crio, parece que cada gesto, cada traço que pretendo fazer, apenas o tenho a uma distância elevada. Para criar uma letra sobre uma tela devo apertar uma pequena tecla de plástico e, como se por milagre, uma letra aparece diante de mim num monitor. Quando desejo que uma linha apareça sobre a tela devo escolher uma ferramenta para isso, clicar e arrastar utilizando um *mouse* até que a linha seja feita numa tela que eu não toquei. E, com o mais simples gesto, tudo que criei pode deixar de existir, se desfazer rapidamente com a ausência de eletricidade alimentadora daquela máquina a minha frente.

Logo, se há algo que diferencia a criação digital daquela no meio impresso, é a mediação entre meu ato e o que crio. São os zeros e uns – como o quer Maria Clara Paixão de Sousa (2009) em seu ensaio *Conceito material de “texto digital”* –, os milhares de códigos que se empilham uns sobre os outros, decifrados e codificados infinitamente através de uma máquina movida por impulsos elétricos que se antepõem a meu movimento, se antepõem ao meu ato de criar, ao que eu pretendo criar. Pois esta é justamente a diferença entre o meio impresso e o digital: a interposição de um código binário entre eu e o que crio. A técnica ou a ferramenta se interpõe no meu ato de criar.

Entretanto, se paro um segundo e penso em todos os materiais que estão sobre a minha mesa, todos os lápis e canetas, e ainda um violão que descansa ali ao lado, posso afirmar que minha vontade criativa não é mediada por eles? Será que em algum deles posso dizer que não há uma mediação entre meu ato e o resultado final? Posso dizer que meu intuito em criar não levaria em consideração o lápis ou o violão que ali repousam? Será mesmo que o que difere o meio digital é realmente o fato de ele ser “mediado” por zeros e uns? Não posso encontrar um ato criativo que não seja “mediado” ou que não tenha no mundo outro elemento que se interponha – se coloque entre o eu que crio e o mundo –, nem mesmo traçar desenhos sobre

a areia com meus próprios dedos está livre de um efeito que não é meu, de uma outra força trabalhando aquilo que crio. Talvez apenas o canto possa escapar a ser mediado por algo, mas ao mesmo tempo ele pode ser considerado uma modulação da minha expressão diária (e poderia ainda dizer o mesmo da dança enquanto uma modulação da minha movimentação espacial, ambas que devem ser aprendidas por mim) (MERLEAU-PONTY, 2006), e o modo que ele se dará ao ouvido dos espectadores não será o mesmo que ouço dentro de meu próprio corpo, cercado por mim mesmo, eles ouvirão o canto com relação a uma distância espacial, rebatido por paredes, alterado pelo espaço que os cerca. Sempre haverá outro fator em jogo, nunca crio com base numa ideia pré-criada em algum lugar do meu *cogito*. Não tenho e transfiro uma criação perfeita e acabada de um *res cogitans* para uma *res extensa* como um ato de mágica. A criação já está no mundo, já está imersa nele e já conta com os utensílios utilizados¹¹.

Se percebo o escrever com um lápis sobre uma folha de papel – algo que costumo fazer – como um ato direto é porque ignoro toda uma gama de fenômenos físico-químicos que ali têm seu lugar. Estou tão acostumado com o ato de traçar que ele já não pode mais me parecer distante. O meu criar ou desenhar sobre uma folha com um lápis apenas será uma “mediação” – algo se interpondo entre meu corpo e o mundo – se olhar aquele ato objetivamente, enquanto duas exterioridades, enquanto dois objetos – tanto o lápis e meu corpo; ignorando que todo movimento que faço sobre o papel é pressuposto pelo meu corpo, ele sabe como ele acontecerá, ou sabe esboçar seu acontecimento, compreende como ele mesmo e o lápis se comportarão e como os traços sairão sobre a folha. Mas todo esse hábito se desmancha se peço entre os dedos um giz de carvão ou um pincel de bambu que nunca manuseei. Existe uma similaridade, logo o ato de traçar não será estranho, porém ele será outro, e aquele utensílio que porto entre meus dedos já me distanciará do que pretendo, borrarei a tinta, farei traços mais grossos ou desproporcionais, ou ainda, não saberei exatamente o que posso conseguir com aquela ferramenta, terei que reaprender meus gestos, reaprender o que pode vir a ser de minha interação com ele.

Acredito que o mesmo ocorre com a criação no meio digital. Meu traçar é o digitar um código, meu movimento sobre o papel, minha escolha de cores e sons, tudo isso acontece por meio de outros caminhos, mas que não são um abismo entre mim e minha criação¹². Esse abismo de mediação só aparece se aquele objeto é estranho para mim, se ele ainda não faz parte de meus movimentos, se ainda não

¹¹ Apontar a criação digital como uma criação que tem seu diferencial na mediação parece remontar àquela clássica visão platônica dos graus de distanciamento de uma criação terrena com relação à Ideia.

¹² O que existe de diferença é o caráter do ato: enquanto antes um gesto físico explícito levava o pincel a traçar aquela linha que desejava, agora isso se dá por meio de um código lógico, mas que não deixa de ser uma técnica a ser aprendida.

consigo ter ele enquanto uma parte do meu corpo. E isso acontecerá com qualquer ferramenta utilizada, até mesmo minha própria voz no canto: não terei domínio dela até aprender suas nuances, aprender como operá-la e utilizá-la para soar do modo que almejo. Dançar, que talvez pareça tão simples para um espectador, se mostrará como um ato de extrema complexidade para aquele que tente a primeira vez. Seu corpo parecerá quase estranho e aqueles movimentos tão simples e precisos, impossíveis.

O código binário de zeros e uns que se interpõe e medeia minha criação representa um invisível que não pode ser utilizado como fonte única para caracterizar uma criação digital. Podem, enquanto descrição técnica, material, objetiva e empiricista, servir para caracterizar o objeto, mas se formos estudar a poesia digital através de uma perspectiva fenomenológica, ou em qualquer outra linha em que a apreensão do fruidor esteja em jogo, essa diferenciação do impresso e do digital por meio de zeros e uns é inviabilizada. Ora, essa mesma mediação ocorre sempre, de uma forma ou de outra. A combinação química que ocorre numa tela com a tinta recém espalhada altera sua cor, sua textura, e o tempo vai lentamente alterá-la; existem eventos ali que não vemos e que alteram o resultado, mudam as cores e a totalidade ali exposta. Existe sempre o jogo entre dois ou mais elementos que se alteram – tudo afeta tudo. O criador se apropria dessas alterações, ele as conhece “instintivamente” através do constante contato com aquele meio/ferramenta/objeto, as alterações entre seu contato e o resultado final passam a ser esperadas e compreendidas de certo modo por ele, não são um fator estranho que vem intervir no processo de interação, são parte do processo de interação.

Um breve exemplo

Permitam-me exemplos de minha própria lavra. O *Sintext* é o que vem a ser chamado de gerador de texto automático, criado pelo português Pedro Barbosa (2010). No caso, um gerador de texto automático faz bem o que o nome diz: gera textos automaticamente no computador com base num código pré-programado. Geralmente isso é feito no código por via da permutação de certa quantidade de variantes. É o caso do *Poetry Creator2*, de Jeff Lewis e Erik Sincoff (2009), que gera um pequeno poema automaticamente (inserindo nele algumas palavras que o leitor pode preencher). Ou também é o caso do *Adolescent Poetry Generator*, criação de Josh Larios (2010) que gera poemas de estilo “adolescente” a partir da permutação de certos elementos num banco de dados. Mas no caso específico do *Sintext*, é o leitor que determina tanto o programa permutativo (a estrutura e como ela será permutada) quanto o léxico a ser permutado (o que será efetivamente permutado). Para fazer isso o leitor deve manipular um código simples (pertencente ao programa)

que basicamente envolve escrever algo denominando caracteres (sejam eles palavras ou frases e por diante) de elementos fixos ou variáveis. Os fixos permanecem na permutação e os variáveis são permutados entre o léxico. Essa estrutura pode ser aplicada de parágrafos variantes até letras variantes dentro de estruturas maiores.

Na primeira tentativa de utilização desse programa, tentei simplesmente criar versos permutáveis (algo como um gerador de centões¹³ automático). Tentei fazer isso por algum tempo, já que havia constantes erros meus na escrita do código que travavam o programa. Eventualmente comecei a olhar os três exemplos de geradores que vêm com o *Sintext* (pode-se acessar três exemplos e seus códigos) e percebi que a complexidade poderia ser aumentada. Dai, comecei a jogar mais com as possibilidades, comecei a tentar permutar ainda coisas dentro de coisas, números em vez de letras e assim por diante, até que consegui perceber o que seriam os resultados de um programa com base no que via do código, ou seja, conseguia escrever já pensando na permutação que ocorreria. De alguma maneira, ela se tornou parte da minha escrita, eu podia criar textos em que os parágrafos eram permutáveis entre si, como também as frases desses parágrafos, e até palavras dentro dessas frases, sem necessidade de ficar elaborando tudo previamente¹⁴.

A minha experiência de aprendizagem com o *Sintext* também é próxima da minha aprendizagem em programação¹⁵. Em 2005, quase nada sabia ainda sobre programação. Através de várias tentativas próprias, e curiosidade acima de tudo, fui aprendendo a lidar com *html*, *flash* e *lingo*. No primeiro momento em que comecei a utilizar o *flash*, vinha ao computador com uma ideia pré-estipulada de uma criação digital a fazer. Com algum tempo de utilização, verificava que aquele programa não poderia fazer aquilo que eu desejava, ou eu não tinha conhecimento suficiente para fazê-lo. Como resultado, eu acabava brincando com o programa e aprendendo seus mecanismos (tanto o *flash* quanto o *Director* tem uma série de efeitos ou possibilidades facilmente acessíveis a quem os usa). Então, acabava fazendo criações que utilizavam estas ferramentas específicas como suas funções básicas, ou seja, deixava que o programa determinasse minhas escolhas na criação. À medida que o tempo passava, comecei a “torcer” as possibilidades dessas ferramentas e me

¹³ Centões são poemas compostos com versos de outros poemas (geralmente de outros autores). Como existem sonetos feito com versos d’*Os Lusíadas*, da *Eneida* e assim por diante.

¹⁴ Algumas dessas minhas brincadeiras mais simples podem ser encontradas no artigo *Algum ritmo algorítmico: possíveis ligações entre o sintext e a poética de E. M. de Melo e Castro* (TAVARES, 2007), como também no artigo de Alckmar Santos (2006), *Algumas Notas Sobre as Leituras de Literárias Digitais*.

¹⁵ É interessante pensar que o *Sintext* traz o código-fonte (apesar de se tratar de um código próprio e relativamente simples) para frente do leitor, ele propõe que este tenha que aprender algo das maquinações do meio digital para poder criar. O *Sintext* acaba indo no caminho oposto dos vários programas *user-friendly* que interpõem uma interface amigável entre o usuário e o seu objeto de trabalho, ele chama o leitor a olhar mais diretamente o nu e o cru do digital.

aprofundar nas possibilidades do programa. Comecei a programar com o programa, e não mais ser guiado por ele. Em algum instante, surgiu o que posso chamar de familiaridade: eu conseguia ler o código olhando-o sem necessitar de referência externa. Não só isso: era possível ver as relações ali programadas, entrever o resultado que aquelas linhas teriam. Ou seja, aquele ato de programar e criar já não me era estranho. O programa passara a ser uma ferramenta minha como o lápis ou o pincel, se tornou comum a meu corpo. Ao criar com aqueles programas, havia incorporado aquelas regras (restrições a criação) a meu campo de possibilidades de expressão¹⁶.

LUTHIER Digital

Esse contato com o meio digital traz à tona uma pergunta simples: até que ponto o autor de um poema digital necessita conhecer o código fonte (no sentido de ter comando sobre sua ferramenta)? Talvez seja adequado pensar em diferentes manifestações artísticas, como por exemplo: um músico não necessariamente é um *luthier*; todavia, ele sabe até certo limite o modo de operar seu instrumento, um violonista sabe como as cordas tencionam e como tirar de seu instrumento o som que deseja, conhece os limites de seu instrumento, conhece aquele objeto, se não os detalhes minuciosos de sua construção, pelo menos os efeitos produzidos por certas interações dele com o objeto. Da mesma forma que um pintor tem um conhecimento necessário das tintas e suas composições e como irão reagir em contato com outras tintas, a tela e ar. Não que ele tenha um conhecimento de sua precisa composição química, mas um conhecimento de como ela opera ou pode vir a operar, e daí sim a química, no sentido de ter conhecimento de que uma tinta é ácida ou que não se mescla com tal outro tipo de tinta e assim por diante (a partir disso, temos a química na medida em que ela é observável ou gera efeitos no ato de pintar).

Um leitor ou ouvinte de música ou literatura não precisa indispensavelmente conhecer a técnica para fruir aquele objeto estético. Se conhecer, melhor ainda, pois terá a capacidade de perceber e fruir outros elementos na obra que não os diretamente perceptíveis. Sabemos que a percepção de um músico com relação a uma melodia tocada é diferente daquela que possui um leigo, e no caso de um poema, poderíamos falar o mesmo daquele que possui uma maior percepção rítmica com relação ao poema que tem uma metrificação.

Por outro lado, uma percepção mais aguçada do objeto não é imprescindível para fruí-lo. Contudo, no momento em que passo para um lugar de criador,

¹⁶ É interessante frisar que a aprendizagem dessa generalidade do código também afetou meu modo de lidar com computadores – entendendo mais a fundo a sua estrutura geral e comportamento possível – e, mais importante, o meu modo de ler os poemas digitais que agora pareciam mais próximos ou mais palpáveis para a interação.

podendo criar ou participar da construção da obra, eu necessito conhecer parte do seu funcionamento para poder fruir. Não que isso queira dizer um conhecimento necessariamente da técnica (apesar de poder ser), mas sim em termos dos elementos em jogo na obra e como podem operar naquela dada situação, suas possibilidades e limitações.

E aí está uma das questões que vem à tona com uma arte participativa no meio digital. Se o leitor agora tem que participar da obra para fruí-la, aproximando-se do lugar de um autor, até o ponto de certos críticos falarem de “escreitor” (BARBOSA, 1996) ou co-autoria, então se torna necessário que o leitor também se aproxime de certas atividades do autor. Porém, devemos pensar que isso não ocorrerá da mesma maneira como ocorre com o autor, pois ao autor ainda são reservados certos privilégios de concepção e de criação (por sua posição privilegiada em ser aquele que decide os passos iniciais da obra) aos quais o leitor não terá acesso exatamente. Logo, não poderia haver um nivelamento da obra como criação de ambos, autor e leitor, mas é necessária uma aproximação do modo de operar do leitor ao do autor – tendo em mente que ambos não são a mesma coisa e que existem diferenças procedurais da possibilidade de cada um com relação à obra.

Existe no ato de criação e de leitura uma confluência, mas elas não são a mesma coisa. O criador tem uma infinidade de possibilidades diante de si, o leitor tem um mundo em miniatura, regido pelo horizonte e vontade (distanciada) do criador. A confluência está no “tornar parte de si”, de aprender as regras e se tornar parte daquilo e tornar aquilo parte de si para poder interagir com aquele mundo.

Torna-se necessário ao leitor se apropriar do funcionamento da obra da mesma forma que o autor se apropriou do funcionamento das ferramentas que utilizou para criar, porém o conhecimento do leitor em relação à técnica que o autor utilizou pode mudar ainda sua apreensão do objeto. Ainda que eu não mexa no código da obra como leitor, o conhecimento da programação (da dada linguagem) pode afetar minha leitura; da mesma forma que um relojoeiro mesmo sem abrir um relógio pode dizer coisas sobre este, prever defeitos, ou entender do que se trata, apenas pelos sons e movimentos dele. Seria um internalização do conhecimento de seu funcionamento que passa a dialogar com as manifestações externas (inseparáveis). Falo de compreender a sua operação externa, olhar o objeto de outra forma e vislumbrar como foi construído, olhar e saber julgar a qualidade dos gráficos utilizados, que tipo de programa foi utilizado, se aquilo foi “pré-renderizado”¹⁷, se ele pode se mover, se alguém o desenhou.

Podemos pensar na capacidade de, ao olhar uma pintura, sabermos o que era possível com aquela tinta, entender melhor a pintura, como um conhecimento do

¹⁷ *Pre-rendered* em inglês, em especial com relação a jogos de vídeo-game, se refere a gráficos.

corpo, não em termos de uma dicotomia, corpo e mente, e sim no sentido de uma sensibilidade ao que é percebido. Isso significa um esboço do que seria possível com meu corpo naquela obra, o que inevitavelmente acarreta a necessidade de uma nova sensibilidade.

Nova sensibilidade

A fabricação de um novo objeto estético, produzido em outro meio, implica o nascimento de uma nova sensibilidade em quem cria e na necessidade de uma nova sensibilidade em quem frui. Essa nova sensibilidade não pode apenas se dar no âmbito distanciado, ela implica um novo modo de construir, outro modo de olhar o mundo, com relação tanto às obras imediatamente produzidas naquela nova sensibilidade, tanto quanto nas que a antecederam. Seria algo como a mudança estrutural causada por um novo conhecimento que acarreta uma alteração no que foi e será experienciado (MERLEAU-PONTY, 2006).

Com a mudança de meios, nasce um novo modo de leitura, outro horizonte do que seja leitura, outras técnicas. Deve haver um acompanhamento técnico, se a escrita digital mudou, a leitura também há de mudar. Se aproximo o autor e o leitor, não posso manter uma interação de primeiro plano. A obra digital é produzida no código (mas não apenas nele, obviamente!); logo, se pretendo me aproximar do autor, tenho também que ir até ele, não permanecendo apenas no que o autor propôs como labirinto¹⁸. É preciso que se faça a tentativa de compreender a construção do labirinto. Não no sentido de saber o código, mas sim como os códigos funcionam, entendendo como a máquina é.

Este tem sido meu modo de operar até o momento, tentar entender as criações no meio digital, lançando-me nele, também me lançando nas criações antigas – experimentais e até mesmo barrocas – para nelas tentar apreender as possibilidades do que podemos chamar poema digital. Mas, ao mesmo tempo, entranhar-me no código e nos antigos poemas não se torna um modo de ler o presente. Na verdade, isso aponta para uma situação de reciprocidade, em que o presente ensina a ler os poemas antigos, ensina a ver neles um funcionamento que não seria possível se não partisse do ponto em que me encontro. Se posso falar de programa, de interação, no meio impresso tanto quanto no digital é porque tentei olhá-los ao mesmo tempo, deixei que o presente me ensinasse a ler os poemas impressos e os digitais, deixei que uma técnica de criação me mostrasse o que havia de tão comum entre ambos, me

¹⁸ Como um rato que, em princípio, teria escolhas dos caminhos a seguir dentro de um labirinto. No entanto, todos foram previstos pelo autor..

fizesse perceber que o digital não representa uma experiência alienígena como algo de uma ficção científica, mas algo presente no mundo e convergente com tantas outras criações que a precederam.

Resposta

Voltemos àquelas duas perguntas postas no início do segundo capítulo. A primeira: por que olhar o código fonte se este se encontra por detrás da obra de tal forma que não é visto ou tocado pelo leitor? Bem, um leitor realmente não precisa conhecer o código, mas existe uma nova sensibilidade que aflora e se este se propuser a conhecer os novos meios que a cercam, tanto o código, quanto a máquina e seus funcionamentos, ele terá uma sensibilidade maior com relação aos objetos ali produzidos e pensados. O crítico precisa conhecer o código, a máquina, o mundo no qual a obra ganha vida, ou ele irá sempre se ater a uma leitura do que é mostrado (o que não é inválido, mas é certamente limitado), estará sempre preso ao que seria uma análise “conteudista” daquele objeto, estará a criticar o que ele não conhece. O criador precisa saber o código, o meio, tudo que ali se insere, precisa se lançar dentro daquele meio, sem se deixar absorver totalmente por ele, ele precisa conhecer nem que seja pra saber o que é possível. Existem programas que permitem criar sem olhar o código (o que seria uma tentativa de dizer que “isso é fácil, qualquer um pode fazer”, tentando afastar o usuário da parte complicada, interpondo interfaces “amigáveis” entre o usuário e o código, ou seja, entre ele e a matéria bruta que permite com que ele crie e faça o que deseja), porém, compreender seu funcionamento implica entrar naquela linguagem, significa conhecer e manipular os elementos.

A segunda pergunta: não será o código um domínio fora da composição explícita da obra, dado que até mesmo muitos autores digitais não programam suas obras, mas recorrem a um programador para a tarefa (evidenciando um afastamento do autor com relação à técnica de composição ou à complexidade que distancia o autor de um conhecimento de uma técnica tão específica)? Bem, por tudo que já foi mostrado não podemos acreditar que o código seja algo fora da composição da obra. O fato de muitos autores digitais recorrerem a programadores para a composição da suas obras não implica uma desvinculação entre o ato de criar e o de programar. Na verdade, isso parece uma espécie de retorno aos tempos renascentistas, em que um ateliê de pintura não estava segregado de uma oficina de engenharia, em que criar uma obra de arte não era considerado uma inspiração pertencente a um gênio fora desse mundo, mas um trabalho próximo ao do artesão ou cientista, um trabalho que necessita de uma técnica que não poderia se especializar. O que temos hoje com o artista

que trabalha ao lado do programador não é um trabalho criativo ao lado de um técnico-científico, e sim a evidência de que a criação artística não foge à ciência nem à técnica. O erro estará em pensar o cientista da computação ou técnico como menos “artístico” do que o autor, de pensar que um poderia existir sem o outro nessa criação que chamamos de poesia digital. Logo, o código ainda é parte da composição da obra, nunca deixou de ser, e não poderia ser de outro modo.

TAVARES, O. G. Considerations regarding the source code of digital poetry. **Revista de Letras**, São Paulo, v.50, n.2, p.447-467, jul. /dez. 2010.

- **ABSTRACT:** *This article has aims on mapping and analyzing aspects of computer science, on how they affect creation, aesthetic reception and operating mode of digital poetry. Specifically, it means to look at the programming language as part of the creation and constitution of the digital literary work, as that which stands behind what is shown to the reader, as a restriction and potentiality of creation, and as a signifier in a construction that has its own materiality as a giver of meaning. And finally, it intends on comprehending up to what point the knowledge of these modes and mediums are necessary for both the creation and reception of the digital literary works.*
- **KEYWORDS:** *Digital literature. Technology. Computer science. Programming Language. Creation. Aesthetic reception.*

Referências

BARBOSA, P. **A ciberliteratura:** criação literária e computador. Lisboa: Edições Cosmos, 1996.

_____. **Sintext.** Disponível em: <<http://www.pedrobarbosa.net/SINTEXT-pagpessoal/SINTEXT.HTM>>. Acesso em: 10 jan. 2010.

BOOTZ, P. The problematic of form transitoire observable, a laboratory for emergent programmed art. **Dichtung-Digital – jounal für digitale ästhetik**, Brown University, v.35, n.1, 2005. Disponível em: <<http://www.brown.edu/Research/dichtung-digital/>>. Acesso em: 03 maio 2010. Não paginado.

GADAMER, H.-G. **Verdade e método:** traços fundamentais de uma hermenêutica filosófica. Rio de Janeiro: Vozes, 1999.

LARIOS, J. **Adolescent poetry generator**. Disponível em: <<http://www.elsewhere.org/hbzpoetry/>>. Acesso em: 02 maio 2009.

LEWIS, J.; SINCOFF, E. **Poetry creator2**. Disponível em: <<http://www-cs-students.stanford.edu/~esincoff/poetry/jpoetry.html>>. Acesso em: 02 maio 2009.

MARINHO, F. C. de C. et al. **Palavrador**. 2006. Disponível em <<http://www.ciclope.art.br/pt/downloads/palavrador.php>>. Acesso em: 27 jan. 2010.

MELO, R.; REIS, P. **Poemas em efe**. Disponível em: <http://po-ex.net/index.php?option=com_content&task=view&id=92&Itemid=35&lang=>>. Acesso em: 10 dez. 2009.

MERLEAU-PONTY, M. **Fenomenologia da percepção**. São Paulo: Martins Fontes, 2006.

OULIPO. **OuLiPo**. Disponível em: <<http://www.ouliipo.net/>>. Acesso em: 14 jun. 2009.

POESIA EXPERIMENTAL PORTUGUESA. **Cadernos e catálogos da poesia experimental portuguesa**: anos 60. Disponível em <<http://po-ex.net>>. Acesso em: 07 maio 2009.

SANTOS, A. L. dos. Algumas notas sobre as leituras de literárias digitais. **O Eixo e a Roda**: revista de literatura brasileira, Belo Horizonte, n.13, v.1, p.171-191, 2006.

SOUSA, M. C. P. de. Conceito material de “texto digital”: um ensaio. **Texto Digital**, Florianópolis, v.5, n.2, dez. 2009. Disponível em: <http://www.textodigital.ufsc.br/num09/mariaclara.htm>. Acesso em: 04 fev. 2010.

TAVARES, O. G. Algum ritmo algorítmico: possíveis ligações entre o sintext e a poética de E. M. de Melo e Castro. **Texto Digital**, Florianópolis, v.3, n.1, jul. 2007. Disponível em: <<http://www.textodigital.ufsc.br/num04/otavio.html>>. Acesso em: 07 set. 2009.

TORRES, R. **Amor de Clarice**. Porto: Universidade Fernando Pessoa. 2005. Disponível em: <<http://telepoesis.net/amorclarice/index.html>>. Acesso em: 14 jun. 2009.

