

Arte de Código Aberto

Doutor Carlos de Oliveira Junior (UFRJ)

INTRODUÇÃO

O código aberto (*open source*) é um impulso em direção à popularização da criação de suas próprias ferramentas digitais. Ao disponibilizar abertamente o código, tem-se a possibilidade de olharmos como a máquina funciona, contribuindo especialmente para um processo de aprendizado baseado no desenvolvimento de projetos.

A arte digital compartilhada na *internet* na década de 1990 e 2000 era produzida majoritariamente com ferramentas de características da programação de Código Fechado, tais como o *Macromedia Flash* e o *Java*, e eram compartilhadas *online* nos *sites* pessoais dos artistas-programadores, o que contribuiu para difundir a prática e construir a comunidade internacional. Nessa mesma década, foram criadas as ferramentas criativas de código aberto com uma semântica privilegiando o processo criativo. Como exemplo, podemos citar o sistema *DBN*, lançado em 1999 por John Maeda, que inspirou a criação do *Processing* em 2001 por Ben Fry e Casey Reas que, por sua vez, inspirou o *Arduino*, criado em 2005 por Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis. Aqui, é possível perceber uma das características fundamentais do formato de criação com código aberto, que é sua capacidade de inspirar possibilidades para além do projeto original. Também nessas décadas foram desenvolvidos conceitos (*Software livre*, *Open source*), protocolos de controle de versões e de colaboração (*SVN*, *GIT*) e licenças (*BSD*, *GPL*, *Copyleft*, *Creative Commons*) que possibilitaram o florescimento do código aberto, processo hoje consolidados na indústria de desenvolvimento do *software*. Para além das ferramentas criativas de código aberto, este artigo tem como objetivo observar a produção artística que é compartilhada a partir do paradigma do código aberto. O sociólogo Manuel Castells vislumbrou uma arte de código aberto da seguinte forma:

A Internet oferece a possibilidade de criação coletiva, interativa, conjunta, por meio de práticas de grupo que permitem a pessoas distantes no espaço pintar, esculpir, desenhar, compor e produzir juntas, em interação e muitas vezes em contradição. O mais das vezes, esses artistas não se conhecem, exceto por sua arte — e isso é tudo que importa. **A arte de fonte aberta é a nova fronteira da criação artística.** Além disso, a abertura da web democratiza verdadeiramente a arte. Websites oferecem o legado da arte, bem como criações em curso, com netizens do mundo inteiro sendo convidados a aprender, a propor e a participar da criação. (CASTELLS, 2001, p. 204, grifo nosso)

Com o surgimento das redes sociais na década de 2010 e das plataformas que permitem compartilhar o código aberto, tornou-se possível observar algumas características dessa arte de código aberto. Antes de explorarmos tais características, é válido entender dois fundamentos que constituem a cultura e a prática, que são respectivamente o código aberto e a programação criativa.

O CÓDIGO ABERTO

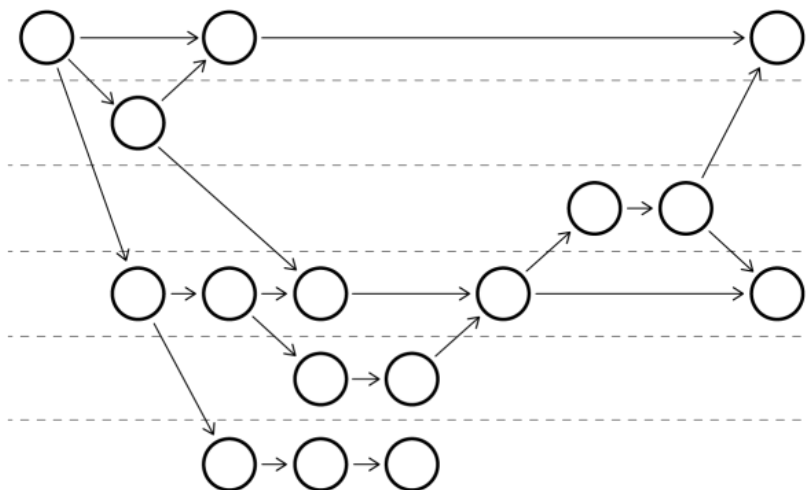
As ferramentas digitais são disseminadas pela *internet* através de uma cultura de compartilhamento que utiliza e mistura estas ferramentas dentro da filosofia do *software* livre, posteriormente desdobrada no código aberto. Um projeto de código aberto não é caracterizado exclusivamente pela disponibilização do código fonte de um programa de computador. Além dessa característica fundamental, a organização *Open Source Initiative* elenca em seu *site* 10 aspectos que definem projetos de código aberto, dos quais destaco os seguintes: livre redistribuição, permitir trabalhos derivados e manter a integridade do autor do código fonte (OPENSOURCE.ORG, 2022). Há um cuidado em manter o projeto de código aberto de maneira que ele continue aberto, o que possibilita construir uma ideia de coletividade que se realiza em uma comunidade que copia. As cópias poderão ser utilizadas, alteradas e melhoradas (num sentido performático do *software*) e, muito comumente, de maneiras não previstas pelo autor. Essas pessoas participam em diferentes níveis de engajamento, reportando erros, comentando ou mesmo sugerindo mudanças no código. A generosidade de quem disponibiliza o código aberto é

recompensada por essa comunidade que pode devolver mais do que recebeu, criando uma dinâmica social que se auto abastece emocional e economicamente.

Para que essas trocas fossem eficientes, no sentido de rapidez e de precisão, foi necessária a criação de um sistema de versionamento de código. Esses sistemas implementam protocolos que estabelecem como as alterações realizadas nas cópias individuais serão combinadas para possibilitar que múltiplas pessoas alterem simultaneamente o mesmo arquivo. O mais popular desses sistemas é o Git, criado em 2005 por Linus Torvalds, mesmo criador do sistema operacional de código aberto Linux.

As contribuições realizadas utilizando o sistema Git ficam armazenadas em um repositório onde é possível navegar pelos arquivos do projeto, assim como observar o histórico de contribuições, verificando quando e quem realizou alterações em cada arquivo. As contribuições são submetidas para os repositórios através de *commits*. Um *commit* contém todas as linhas adicionadas e apagadas dos arquivos alterados, assim como uma pessoa autora e uma título que pode ser utilizada para descrever as alterações. Dependendo do nível de permissões de escrita que a pessoa tem num repositório, um *commit* pode ser automaticamente mesclado (no inglês, *merged*) ou precisará passar por uma revisão da pessoa mantenedora do repositório através de um pedido de revisão chamado *pull-request*. Cada pessoa pode realizar bifurcações (*forks*) que funcionam como uma cópia do repositório original. Na bifurcação, a pessoa tem controle total dos arquivos e pode realizar alterações sem precisar interagir com o sistema. Se for do desejo da pessoa enviar as alterações ao repositório original, esse procedimento se dará via *Pull-request*. O processo colaborativo é descrito no diagrama abaixo (OLIVEIRA JUNIOR, 2020, p. 70-71), no qual cada linha representa uma cópia (*Fork*) daquele repositório e cada círculo representa uma contribuição (*commit*):

Figura 1 - Representação da colaboração presente nos projetos de código aberto.



(OLIVEIRA JUNIOR, 2021, p. 70)

O círculo no canto superior direito do diagrama representa a primeira versão do projeto. O fluxo de contribuições que se seguem não necessariamente precisa voltar ao projeto original; eles podem se abrir para outros projetos com diferentes usos e finalidades. As possibilidades de utilização desse processo colaborativo não se restringem aos projetos de *software*. Projetos que tenham em sua essência dados textuais podem se beneficiar dessa dinâmica. Como se vê:

O modelo colaborativo pode ser observado no livro *The Book of Shaders*, escrito por Patricio Gonzalez Vivo e Jen Lowe, disponibilizado em código aberto. Considere que ao obter a cópia do livro, um leitor possa encontrar um erro de digitação, corrigi-lo e enviar novamente a correção ao autor. O autor pode ou não incorporar essa alteração no livro original. Algumas páginas adiante, o leitor complementa um parágrafo com algo que julgue relevante, mas que não se aplica a outros, assim fazendo uma personalização. É possível guardar a alteração no livro sem obrigatoriamente enviar ao livro original. Ao longo do tempo, outras pessoas fazem contribuições neste livro, de modo que é possível sempre atualizar a cópia que se tem sem necessariamente perder as alterações já feitas. Um caso recorrente desse tipo de colaboração são as traduções. Em 11/07/2020, o *The Book of Shaders* havia sido traduzido para dez idiomas. (OLIVEIRA JUNIOR, 2021, p. 70-71)

A utilização do código aberto como estratégia para desenvolvimento de projetos foi amplamente adotada pela indústria de *software*. As maiores empresas de tecnologia (Amazon, Apple, Google, Meta - antigo Facebook- e Microsoft) possuem diversos projetos de código aberto mantidos por seus funcionários. Essa estratégia funciona não apenas para receber a contribuição resultante da inteligência coletiva via *internet*, como também ajuda a popularizar a adesão do projeto de *software* para toda a comunidade, não apenas divulgando, mas criando vínculos que resultam em grande comprometimento das pessoas com os projetos. É possível observar esse nível excepcional de engajamento em casos em que os projetos encerrados pelas empresas foram continuados pela própria comunidade.

O sistema de criação com código aberto é também uma mudança do paradigma econômico, uma vez que a aritmética não é suficiente para explicar a sustentabilidade do modelo econômico, recebendo mais do que se dá. Na opinião do sociólogo e defensor do *Software Livre* Sérgio Amadeu:

O bem informacional não sofre a escassez típica dos bens materiais. Além disso, o bem informacional não vive o desgaste quando é utilizado. Estas principais características geram enorme dificuldade de transformar o bem informacional em um bem privado, ou seja, a apropriação privada da informação, tenha a forma que tiver, é extremamente difícil. (SILVEIRA, 2007, p. 34, grifo nosso)

A não-escassez e o não-desgaste da cópia no meio digital são as características de base que impõem a um paradigma de trocas dos bens informacionais de forma livre. Essa dinâmica pode ser observada em áreas do conhecimento que criam produtos e serviços a partir da criatividade, o que se convencionou chamar de economia criativa, e compreende tanto a indústria do *software* quanto o mercado da arte. Tendo o código como origem na indústria do *software*, o encontro com a arte se dá através da abordagem experimental, chamada de programação criativa.

PROGRAMAÇÃO CRIATIVA

A ideia de utilizar o computador como meio para produção artística esteve presente desde os primeiros experimentos artísticos criados nas décadas de 1950 e de 1960 pelos pioneiros da arte feita em código: Frieder Nake, Michael Noll, Vera Molnár, Manfred Mohr e Georg Nees. No Brasil, tem-se o pioneirismo de Waldemar Cordeiro. A arte feita em código utiliza o computador como meio criativo, expressivo e crítico. O termo que ajudou a definir a abordagem experimental praticada por aqueles artistas-programadores foi cunhado por John Maeda: a programação criativa “não é uma ferramenta; é uma nova maneira de se expressar” (MAEDA, 2004, p. 113, tradução nossa).

A programação criativa é uma prática utilizada para criar experimentos audiovisuais utilizando o código. Adota uma abordagem experimental, incorporando as descobertas encontradas durante o processo criativo. No contexto das Ciências Exatas, a computação é caracterizada pelo seu determinismo, em que se espera alcançar resultados idênticos dadas as mesmas condições. Assim, a programação irá definir as regras que determinarão o comportamento de um programa de computador. A programação criativa admite a imprevisibilidade do processo criativo e joga com as possibilidades de resultados. A abordagem experimental da programação criativa é característica da Arte, como reconhecido por Guto Nóbrega durante as aulas do curso de pós-graduação em Artes Visuais da UFRJ (PPGAV): “A Arte é um jogo em que as regras são construídas no jogar”. Não necessariamente há um compromisso com um objetivo inicialmente traçado; existe uma abertura para as descobertas do processo, o que pode torná-lo mais leve, favorecendo uma experiência de aprendizado e de experimentação em programação de computadores.

Em *Aesthetic Programming* (SOON e COX, 2020) é apresentada uma visão crítica sobre o avanço do *software* na sociedade, além da utilização da programação criativa como maneira de abordar subjetividades da tecnologia numa perspectiva tecno-feminista e decolonialista. Isso nos encoraja a desafiar a separação dos domínios do *software* e da arte.

Podemos encontrar alguma evidência disso [forças produtivas que são liberadas, mas nunca totalmente contidos pela economia capitalista] nas energias sociais do movimento do código livre e aberto, por exemplo, onde a compensação opera no nível da troca social. Essa afirmação serve então para desviar a atenção dos esforços do indivíduo para as relações sociais. A política disso é especialmente importante se quisermos desenvolver uma posição diferente da lógica de “capturar tudo” e buscar interpretações mais positivas e esperançosas. (SOON e COX, 2020, p. 116, tradução nossa)

A programação criativa se revela como um dos caminhos possíveis para adentrar e alterar simultaneamente nos domínios do *software* e da arte. Durante a participação de um *workshop* com a autora Winnie Soon, houve a oportunidade de perguntar como ela relacionaria o Pensamento Crítico com a programação criativa, a qual ela responde: “Imagine programar não apenas para se expressar, através da programação podemos pensar como as coisas estão estruturadas. (...) É uma ferramenta para o questionamento” (SOON, 2021, tradução nossa). De imediato, houve o reconhecimento de certa continuidade da definição de John Maeda. Em ambas as definições também há a presença da palavra Ferramenta. Maeda parecia já estar respondendo a uma discussão quando dizia “não é uma ferramenta”, ou seja, a programação não deve ser vista apenas pela perspectiva técnica. A programação enquanto técnica é uma poderosa maneira de produzir ferramentas digitais ou ferramentas físicas que dialogam com um *software*. Mas somente se usada criticamente é possível repensar as dinâmicas do *software* na sociedade; de outra maneira, o *software* continuaria fazendo a manutenção de estruturas de poder já estabelecidas.

ARTE DE CÓDIGO ABERTO

Uma arte de código Aberto apresenta junto do resultado áudio e/ou visual os códigos necessários para reproduzir o trabalho. O código não necessariamente precisa fazer parte do trabalho ou ter algum sentido estético no código. Ao disponibilizar o código, há uma alteração significativa na forma como o trabalho se apresenta e vai interagir com as pessoas. O código disponibilizado viabiliza que outras pessoas possam observar as estratégias criativas adotadas e

aprender com elas. A possibilidade de olhar o código de uma Arte de código aberto acrescenta à experiência estética a possibilidade de uma experiência didática, pois permite que outras pessoas possam investigar o processo, adentrar à caixa preta, entender e alterar seu funcionamento. Como explicitado pela *Open Source Initiative*, “código aberto não significa apenas acesso ao código fonte” (OPENSOURCE.ORG, 2022, tradução nossa), sendo necessário prezar pela livre distribuição e pelo direito do autor.

Ao observar as variações de um trabalho de arte de código aberto realizado por outras pessoas, é possível surpreender-se com as contribuições que alteram significativamente o resultado do trabalho, afinal, “uma idéia, ao ser criada, logo desprende-se de seu autor e retorna a sua base comum que é coletiva, social” (SILVEIRA, 2007, p. 34). Sendo alterações de diferentes níveis, propomos a seguinte classificação inspirada pelas licenças do *Creative Commons*:

Nível 0 - Cópia idêntica, o código não é alterado, apenas copiado.

Nível 1 - Alteram-se as propriedades, mantendo-se a lógica de funcionamento.

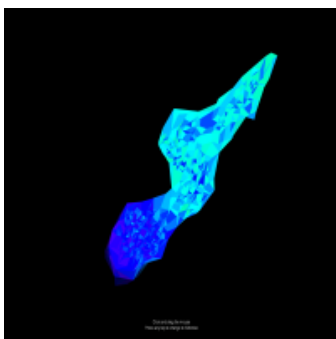
Nível 2 - Altera-se a lógica de funcionamento.

Copiar é uma ação natural utilizada como uma estratégia para o aprendizado. A fundamentação, o raciocínio e a reflexão de aprendizagem são outras fases do processo de aprendizado. Na escola, quando iniciamos a jornada do aprendizado, copiamos repetidamente algo ainda sem entender o problema ou a solução para apenas posteriormente nos tornarmos aptos a aplicar a solução para problemas semelhantes, até que seja então possível expandir a solução para novos problemas. Também aprendemos imitando o modo de fazer do outro.

O mesmo processo experimental da arte que deseja descobrir novos caminhos, novas formas de fazer, de pensar e de existir faz parecer que há uma negação pela cópia, pois haveria um compromisso com a originalidade. Mas admitir na estruturação de um trabalho original que ele é também resultado da contribuição de outros trabalhos torna o processo criativo mais transparente e legitima o que ele é: resultado de uma complexa teia de inspirações, experiências e conhecimentos compartilhados.

EXPERIMENTAÇÃO

Foi realizado um esforço inicial para compreender a dimensão das interações que acontecem com uma arte de código aberto. Esse processo se inicia a partir da possibilidade de explorar a API do *OpenProcessing*. Uma API determina um padrão de comunicação com um sistema e permite consultar ou submeter dados. O *OpenProcessing* é uma plataforma digital que funciona como uma rede social e nela é possível compartilhar apenas arte feita em código. As principais interações disponíveis pela plataforma permitem criar *forks*, comentar e curtir o projeto de outras pessoas. Um projeto criado na plataforma tem a opção de esconder o código, mas tal funcionalidade é pouco utilizada. Na fase exploratória, foi possível identificar que a API do *OpenProcessing* permite resgatar os dados de um projeto e todos os dados derivados das interações. Em seguida, criou-se um *software* de raspagem de dados, o que automatiza o processo de consulta dos projetos mais curtidos da plataforma. O *software* criado foi disponibilizado em código aberto no seguinte endereço: <https://github.com/Vamoss/OpenProcessing-Forks>. Foram coletados os dados dos 100 projetos mais curtidos, porém trabalhamos os dados dos cinco primeiros, citados a seguir:



Frozen Brush
Por Jason Labbe
Id: 413567
curadorias: 3
forks: 303
curtidas: 1946



Messy Curve Draw
Por 郑越升 Sen
Id: 486307
curadorias: 5
forks: 864
curtidas: 1932



Duel (a shooter
game)

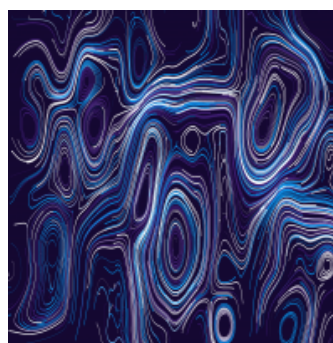
Por FAL

Id: 453716

curadorias: 1

forks: 358

curtidas: 1694



perlin noise

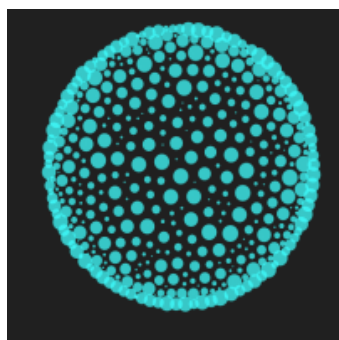
Por Yasai

Id: 494102

curadorias: 3

forks: 320

curtidas: 1418



Wobbly swarm

Por Konstantin

Makhmutov

Id: 492096

curadorias: 2

forks: 283

curtidas: 1304

Frozen Brush, o primeiro trabalho mais curtido da plataforma *OpenProcessing*, foi criado por Jazon Labbe, artista com experiência em efeitos especiais, desenvolvimento de jogos e de animações 3D e que trabalhou em filmes como *Avatar* (2009) e *Vingadores: Era de Ultron* (2015). Ao colocar a URL do trabalho no buscador Google, aparecem dentre os primeiros resultados duas influências desse trabalho para outros estudantes de arte digital. *Frozen Brush* é citado como a principal referência do trabalho de conclusão de curso do então estudante da Universidade de Nova York Shahriar Sadi, como documentado no *blog* (NYU, 2018). A influência do *Frozen Brush* é notável até no resultado da instalação artística desenvolvida por Shahriar. Outro resultado encontrado é uma publicação da então estudante da Universidade de Carnegie Mellon Yuvian, na qual ela relata:

Antes de me deparar com este projeto, eu tinha acabado de começar a experimentar com Processing e P5, brincando com blocos e formas simples. Foi só depois de ver este projeto e

outros semelhantes que percebi o quão dinâmicos, inteligentes, artísticos e interativos as obras de arte computacionais poderiam ser. Admiro as cores neste esboço e a fluidez do movimento do "gelo" à medida que o mouse interage com o código. (GOLANCOURSES.NET, 2018)

A partir do tratamento dos dados, buscou-se uma visualidade que traduzisse a organicidade das interações de uma forma poética sem perder a dimensão informacional da visualização de dados. Foi utilizada a espiral de Fibonacci para recriar a geometria que se assemelha às mesmas estruturas vistas em flores, como no girassol. Cada ponto é uma pessoa que interagiu com o projeto e cada cor representa um tipo de interação; uma pessoa pode ter realizado mais de um tipo de interação e, assim, o ponto pode possuir mais de uma cor.

Visualização das interações de projetos no site OpenProcessing:

Figura 2 - projeto Frozen Brush.

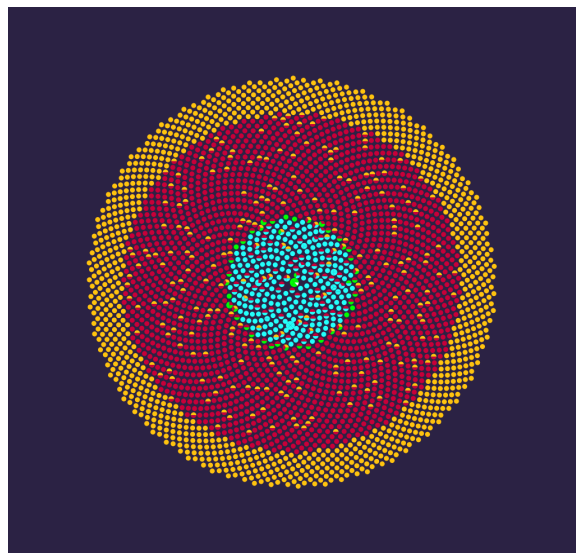


Figura 3 - projeto Messy Curve Draw.

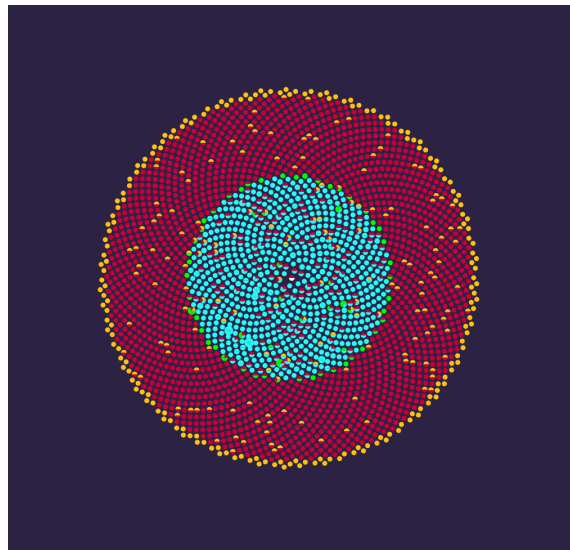


Figura 4 - projeto Duel.

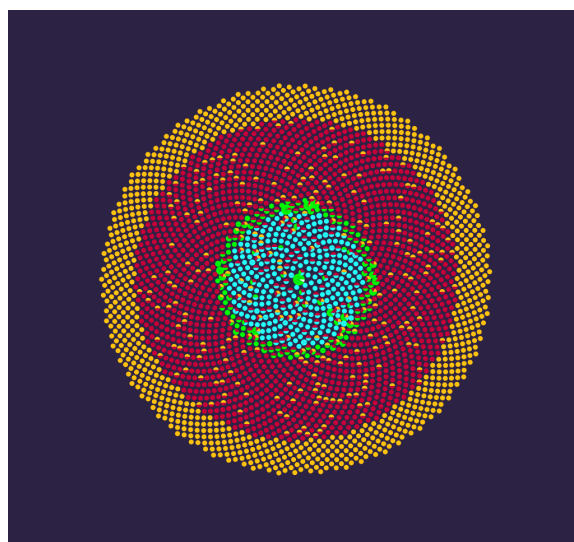


Figura 5 - projeto perlin noise.

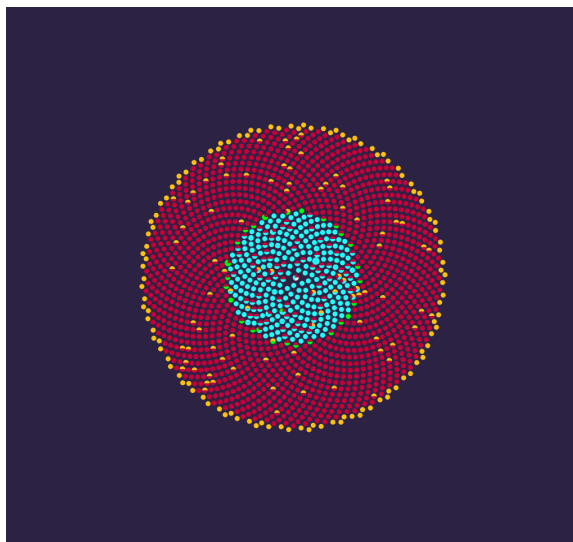
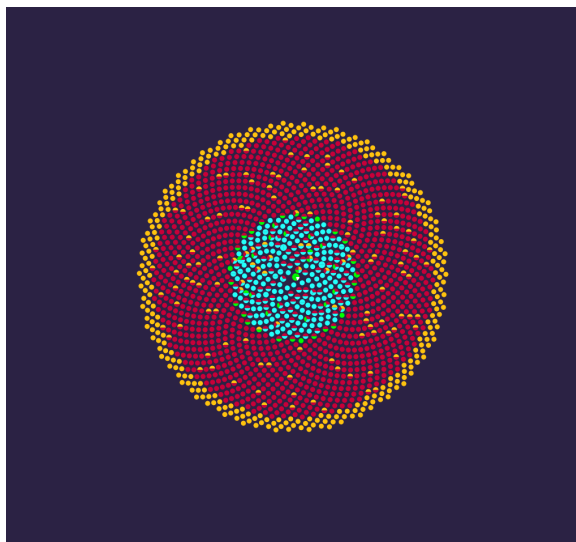


Figura 6 - projeto Wobbly swarm.



Legenda de cores:

- autor
- pessoas que comentaram
- pessoas que seguem o autor

- pessoas que copiaram
- pessoas que curtiram

As interações com os projetos funcionam como polinizadores em que cada interação torna-se o pólen que expande o projeto para além das fronteiras alcançáveis pelo autor. A segmentação de cores nos discos de cada flor evidencia que cada pessoa interage de forma diferente, havendo maior predominância de pessoas que interagiram apenas uma vez. A maior parte das interações partiram de pessoas que não seguem o autor, o que indica a possibilidade de que essas pessoas tenham tido contato com o trabalho de forma orgânica, ou seja, para além da rede do autor. Todos os projetos selecionados foram amplamente copiados. Caberia avaliar numa fase posterior da pesquisa os diferentes níveis de cópia ou a criação de visualização de trabalhos que considerassem outros parâmetros de seleção para além da quantidade de curtidas.

CONSIDERAÇÕES FINAIS

O código aberto altera as formas de se relacionar com o trabalho de arte ao possibilitar que outras pessoas criem variações do trabalho original, difundindo a arte feita em código dentro do paradigma da colaboração. Criar um trabalho de arte nesse contexto é estar aberto a uma vida própria do trabalho, que estará em contato com uma comunidade ávida por estudá-lo e alterá-lo. A vida de um projeto de arte de código aberto se dá através de um diálogo com as pessoas, não apenas através da cópia; estamos falando, então, da construção de significados e caminhos criativos capazes de extrapolar o que fora previsto originalmente. Se na natureza as formas de vida se adaptam ao ambiente, num trabalho de arte de código aberto a vida se dá através de um diálogo com as pessoas que irão olhar, comentar, curtir, compartilhar, usar, brincar, copiar ou mesmo alterar tal trabalho e, assim, difundindo-o nas redes. Essa vida não se restringe ao digital, podendo transbordar para o papel, telas, quadros, instalações, protótipos, dentre outros projetos e, indiretamente, em um consciente cultural, inspirando e se conectando a outras criações. A consequência desse processo vital, nos mundos digital e orgânico, é uma crescente relação com a

materialidade do código, aguçada pela curiosidade, o que promove a familiarização com as engrenagens do *software*. Como resultado, tem-se o questionamento do *software* não apenas no âmbito crítico, como também no âmbito prático, produzindo dentro de uma lógica de generosidade uma vida que se estabelece ao possibilitar que outras se manifestem.

REFERÊNCIAS

CASTELLS, Manuel. *A galáxia da Internet: reflexões sobre a Internet, os negócios e a sociedade*. Trad. Maria Luiza X. de A. Borges. Rio de Janeiro: Jorge Zahar, 2003.

CATLOW, Ruth; GARRETT, Marc, Collaboration and Freedom – The World of Free and Open Source Art. 2011. Website do P2PFoundation. Disponível em: <https://wiki.p2pfoundation.net/World_of_Free_and_Open_Source_Art>. Acesso em: <21/06/2022>

GOLANCOURSES.NET, 2018. *INTERACTIVITY & COMPUTATION*. Disponível em: <http://golancourses.net/2018_60212f/yuvian/09/06/yuvian-lookingoutwards01/>. Acesso em: <21/06/2022>

MAEDA, John. *Creative Code*. High Holborn, Londres: Thames & Hudson, 2004.

NYU, 2018. *Interactive Installations*. Disponível em: <<https://wp.nyu.edu/shahriarinteractive/final-project/>>. Acesso em: <22/06/2022>

OLIVEIRA JUNIOR, Carlos. *Encontros digitais: a construção de uma plataforma digital para comunidade de programação criativa brasileira*. Orientador: Leonardo Marques de Abreu. 2021. 91 f. Dissertação (Mestrado em Economia Criativa). Escola Superior de Propaganda e Marketing, Rio de Janeiro, 2021. Disponível em: <<http://tede2.espm.br/handle/tede/559>>. Acesso em: 6 abr. 2022.

OPENSOURCE.ORG, 2022. *The Open Source Definition (Annotated)*. Disponível em: <<https://opensource.org/docs/definition.php>> Acesso em: <22/06/2022>

SILVEIRA, Sérgio Amadeu et al. A comunicação digital e a construção dos commons: redes virais, espectro aberto e as novas possibilidades de regulação. São Paulo: Fundação Perseu Abramo, 2007.

SOON, Winnie; COX, Geoff. *Aesthetic Programming: A Handbook of Software Studies. Liquid Books*. 2020. Disponível em: <<http://aesthetic-programming.net/>>. Acesso em: 5 mai. 2021.

SOON, Winnie. 2021. Screen Walk with Winnie Soon. *YouTube*. Disponível em: <<https://youtu.be/izNo4iqBlmQ?t=4513>> Acesso em: <22/06/2022>

Como citar este texto:

JUNIOR, Carlos O. Arte de Código Aberto. In: CONGRESSO INTERNACIONAL DE ARTE, CIÊNCIA E TECNOLOGIA e SEMINÁRIO DE ARTES DIGITAIS, 7, 2022, Belo Horizonte. *Anais do 7º Congresso Internacional de Arte, Ciência e Tecnologia e Seminário de Artes Digitais*. Belo Horizonte: EdUEMG, 2022. ISSN: 2674-7847. p. 577-592.