

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Jogos educacionais para ensino em Sistemas Operacionais

Matheus dos Santos Luccas



São Carlos – SP

Jogos educacionais para ensino em Sistemas Operacionais

Matheus dos Santos Luccas

Orientadora: Profa. Dra. Kalinka Regina Lucas Jaquie Castelo Branco

Monografia final de conclusão de curso apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação.

Área de Concentração: Ensino de Computação para nível superior

USP – São Carlos
Junho de 2019

AGRADECIMENTOS

Esse trabalho somente foi possível graças ao apoio de diversas pessoas. Apresento os meus agradecimentos a professora Kalinka Regina Lucas Jaquie Castelo Branco, que apoiou o projeto idealizado a partir do trabalho final da disciplina de Sistemas Operacionais I lecionada por ela no segundo semestre de 2018, além de aceitar ser orientadora deste projeto.

Os meus agradecimentos também ao professor Júlio Cezar Estrela por ter se disponibilizado a apresentar o trabalho aqui feito aos alunos de Sistemas Operacionais I lecionada por ele no primeiro semestre de 2019, o que possibilitou reunir informações indispensáveis para este projeto. Além de prover a ponte entre este trabalho e os alunos, o professor Estrela também auxiliou e deu dicas que influenciaram nas decisões tomadas nesse processo.

Também agradeço a todos os alunos que testaram o trabalho, responderam o formulário com seriedade, deram sugestões e apontaram para os pontos importantes.

Por fim, agradeço ao amigo Marcos Fernando Geromini por ter se disponibilizado para me ajudar com algumas dicas sobre a escrita deste documento e desenvolvimento do trabalho, e para todos que me ajudaram de alguma outra forma.

Sobre o meu estudo na graduação, ele também não seria possível sem a disposição de diversas pessoas. Devo agradecer aos apoios de meus pais Benedito Osmar Luccas e Elizete Tânia Siqueira dos Santos Luccas, também dos meus irmãos Thiago dos Santos Luccas, Luis Felipe dos Santos Luccas e Maria Julia dos Santos Luccas, aos professores da USP. Agradeço também pelo auxílio financeiro dado por James Carlos Siqueira dos Santos, Benedita Aparecida Siqueira dos Santos e Sonice Donizetti Luccas, pela confiança provida por Pedro Orival Luccas, que me inspirou ao processo de transferência para o curso de Engenharia de Computação, pela ajuda incomensurável de Sirlei Aparecida Lucas que ao longo de toda a vida foi como uma segunda mãe para mim, sem ela eu não estaria nem na metade de onde cheguei, que ela descanse em paz.

A todas as pessoas que me ajudaram de outras maneiras.

RESUMO

LUCCAS, M. L.. **Jogos educacionais para ensino em Sistemas Operacionais**. 2019. 68 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Este projeto iniciou-se com base nos desafios do ensino superior de computação, que possuem disciplinas com resumos e conceitos lógicos que, por vezes, possuem teor pouco intuitivo. Desta forma, e uma vez apresentado os conceitos de aprendizagem baseados em jogos, termo que será explicado neste documento, o propósito deste projeto é o desenvolvimento de jogos educativos para o ensino superior, mais especificamente para a disciplina de Sistemas Operacionais.

Neste projeto três jogos foram desenvolvidos com base na disciplina de Sistemas Operacionais ministrados em cursos superiores de informática. São eles: o "Threadman", um jogo de plataforma baseado em tópicos como *thread* e gerenciamento de memória, o "Escalonando", um jogo de quebra cabeça que explora conceitos de escalonamento de processos e o "Race Condition", um jogo de lógica baseado na comunicação entre processos para acesso de regiões críticas.

Para o desenvolvimento e adequação dos jogos, visando um material de apoio ao estudo qualitativo, pesquisas foram realizadas e projetos de aprendizado baseado em jogos foram estudados.

Com o intuito de trabalhar corretamente com a disciplina de Sistemas Operacionais, todos os tópicos chaves que foram tratados nesses jogos foram revistos e resumidos neste documento. Também foi registrado nesse documento os motivos por trás das decisões sobre o estilo visual, efeitos e trilhas sonoras, bem como a mecânica dos jogos em si.

Finalmente, para possibilitar a obtenção de algumas conclusões, os jogos foram apresentados a alunos que estavam cursando a disciplina de Sistemas Operacionais I no primeiro semestre de 2019, que responderam a um formulário desenvolvido para analisar a eficiência do ensino. Por meio do estudo realizado foi possível observar que os jogos contribuíram para a motivação dos alunos com relação ao estudo dos conteúdos apresentados na disciplina.

Palavras-chave: Sistemas Operacionais, Aprendizado baseado em jogos, Ensino de computação, jogos educacionais, Ensino superior.

ABSTRACT

LUCCAS, M. L.. **Jogos educacionais para ensino em Sistemas Operacionais**. 2019. 68 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

This project began based on the challenges in higher computer education that have subjects with abstracts and logical concepts that sometimes have a little intuitive content. In this way and one time introduced to game-based learning concepts, term that will be explained in this document, the purpose from this project is the development of educational games for higher education.

In the project discussed here, three games has been developed based on the subject of Operational Systems taught in higher computer courses. They are "Threadman", a platform game based on topics such as thread and memory management, "Escalonando", a puzzle game that explore process scheduling concepts, and "Race Condition", a logic game based on inter-process communication for access to critical regions.

To the development and the suitability of the games, seeking by a quality study support tool, searches were done and game-based learning projects were studied.

And seeking to work correctly with the subjects from Operational System, every key topics that was use in the games have been taken up and reviewed in this document. Also has been recorded in this document the motives behind the decisions about visual style, effects and soundtracks, as well as the mechanics of the games themselves.

At the end, to make possible to obtain some conclusion, the games were show to students that were studing these subject in the first half of 2019, those student answered one form built to analyze the teaching efficience.

Key-words: Operational System, game-based learning, Computer education, Educational games, Higher education.

LISTA DE ILUSTRAÇÕES

Figura 1 – Interface da ferramenta Unity no Linux	18
Figura 2 – Interface da ferramenta LMMS no Linux	19
Figura 3 – Interface da ferramenta GIMP no Linux	19
Figura 4 – Interface do Google Forms no navegador Mozilla Firefox do Linux	20
Figura 5 – Tela de menu dos jogos.	25
Figura 6 – Tutorial interativo que proporciona um contexto para o jogador.	25
Figura 7 – Cena do jogo <i>Threadman</i> : respeite os semáforos e resgate a informação.	26
Figura 8 – Cena do jogo <i>Threadman</i> : o jogador controla o bloco branco com as setas direcionais. Na região verde, contando com o branco, só poderá haver três blocos.	27
Figura 9 – Cena do jogo <i>Threadman</i> : o jogador precisa coletar o bloco de pontos, representado pelo retângulo de faixa azul, mas o bloco branco não pode entrar na região amarela, pois lá já estão presentes dois blocos rosas.	28
Figura 10 – Cena do jogo <i>Threadman</i> : após recolher o bloco de informação e retornar até a posição inicial, o jogador deverá gerenciar o espaço que ele tem disponível nesses retângulos para depositar a informação e pontuar.	28
Figura 11 – Cena do jogo do Escalonando: escalone os processos e mantenha as filas abaixo da faixa.	30
Figura 12 – Cena do jogo Escalonando: o jogador tem que montar a sua estratégia para manter as filas abaixo da faixa limitadora.	31
Figura 13 – Cena do jogo do Race Condition: organize para que todos os processos passem pela região crítica, apenas um por vez.	33
Figura 14 – Mudança de cores dos blocos que ficam em espera. O jogador não deverá permitir que um bloco chegue a ficar da cor preta, ou a partida se encerra	33
Figura 15 – Cena do jogo <i>Race Condition</i> : no modo variáveis <i>lock</i> , nenhum bloco entrará na região enquanto ela estiver vermelha.	34
Figura 16 – Cena do jogo <i>Race Condition</i> : no modo <i>sleep/wakeup</i> , o jogador pode clicar nos blocos para fazê-los pararem e garantir que somente um acesse a região por vez	35
Figura 17 – Cena do jogo <i>Race Condition</i> : no modo <i>strict alternation</i> , o bloco só entra na região se a parte verde estiver virada para ele	35
Figura 18 – Informações sobre os anos de ingressos e cursos de cada aluno que participou da pesquisa.	39

Figura 19 – Opinião dos alunos em torno da questão 1.	40
Figura 20 – Opinião dos alunos em torno da questão 2.	40
Figura 21 – Opinião dos alunos em torno da questão 3.	41
Figura 22 – Opinião dos alunos em torno da questão 4.	41
Figura 23 – Opinião dos alunos em torno da questão 5.	42
Figura 24 – Opinião dos alunos em torno da questão 6.	42
Figura 25 – Associações feitas pelos alunos dos jogos com o tópico "Algoritmos de escalonamento".	43
Figura 26 – Associações feitas pelos alunos dos jogos com o tópico "Arquivos".	44
Figura 27 – Associações feitas pelos alunos dos jogos com o tópico "Barramento".	44
Figura 28 – Associações feitas pelos alunos dos jogos com o tópico "Chamadas de sistemas".	45
Figura 29 – Associações feitas pelos alunos dos jogos com o tópico "Comunicação entre processos".	46
Figura 30 – Associações feitas pelos alunos dos jogos com o tópico "Condição de corrida".	46
Figura 31 – Associações feitas pelos alunos dos jogos com o tópico "Deadlock".	47
Figura 32 – Associações feitas pelos alunos dos jogos com o tópico "Discos"	47
Figura 33 – Associações feitas pelos alunos dos jogos com o tópico "Entrada e saída". .	48
Figura 34 – Associações feitas pelos alunos dos jogos com o tópico "Escalonamento de processos".	49
Figura 35 – Associações feitas pelos alunos dos jogos com o tópico "Espera ocupada". .	49
Figura 36 – Associações feitas pelos alunos dos jogos com o tópico "Estrita alternância".	50
Figura 37 – Associações feitas pelos alunos dos jogos com o tópico "Exclusão mútua". .	50
Figura 38 – Associações feitas pelos alunos dos jogos com o tópico "Gerenciamento de memória".	51
Figura 39 – Associações feitas pelos alunos dos jogos com o tópico "Monitores".	52
Figura 40 – Associações feitas pelos alunos dos jogos com o tópico "Mutexes".	52
Figura 41 – Associações feitas pelos alunos dos jogos com o tópico "Paginação".	53
Figura 42 – Associações feitas pelos alunos dos jogos com o tópico "Preempção".	54
Figura 43 – Associações feitas pelos alunos dos jogos com o tópico "Preempção".	54
Figura 44 – Associações feitas pelos alunos dos jogos com o tópico "Região crítica". . .	55
Figura 45 – Associações feitas pelos alunos dos jogos com o tópico "Semáforos".	55
Figura 46 – Associações feitas pelos alunos dos jogos com o tópico "Threads".	56
Figura 47 – Associações feitas pelos alunos dos jogos com o tópico "Variáveis de trabalho".	57

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Motivação e Contextualização	13
1.2	Objetivos	14
1.3	Organização	14
2	MÉTODOS, TÉCNICAS E TECNOLOGIAS UTILIZADAS	15
2.1	Considerações iniciais	15
2.2	A disciplina de Sistemas Operacionais	15
2.3	As ferramentas utilizadas	17
2.3.1	<i>A ferramenta Unity</i>	18
2.3.2	<i>A ferramenta LMMS</i>	18
2.3.3	<i>A ferramenta GIMP</i>	18
2.3.4	<i>A ferramenta Google Forms</i>	19
2.4	Considerações finais	20
3	DESENVOLVIMENTO	23
3.1	Considerações iniciais	23
3.2	Como aprender com os jogos	23
3.3	Os Jogos	24
3.3.1	<i>Dinamismo do menu e do jogo</i>	24
3.3.2	<i>Os tutoriais</i>	24
3.4	<i>Threadman, o jogo das threads</i>	26
3.4.1	<i>Objetivo do jogador</i>	26
3.4.2	<i>Relacionando a disciplina com o jogo Threadman</i>	29
3.5	<i>Escalonando, o jogo do escalonador</i>	29
3.5.1	<i>Objetivo do jogador</i>	29
3.5.2	<i>Relacionando a disciplina com o jogo Escalonando</i>	31
3.6	<i>Race Condition, o jogo das comunicações</i>	32
3.6.1	<i>Objetivo do jogador</i>	33
3.6.2	<i>Relacionando a disciplina com o jogo Race Condition</i>	36
3.7	Trilha sonora	37
3.8	Visual	38
3.9	Estudo de eficiência	38

3.10	Resultados obtidos	39
<i>3.10.1</i>	<i>Pesquisa de opinião</i>	<i>40</i>
<i>3.10.2</i>	<i>Perguntas de associações</i>	<i>42</i>
<i>3.10.3</i>	<i>Perguntas dissertativas</i>	<i>56</i>
<i>3.10.4</i>	<i>Pesquisa de sugestões</i>	<i>60</i>
3.11	Considerações finais	60
4	CONCLUSÃO	63
4.1	Contribuições	63
4.2	Relacionamento entre o Curso e o Projeto	63
4.3	Considerações sobre o Curso de Graduação	64
4.4	Trabalhos futuros	64
REFERÊNCIAS		67

Capítulo 1

INTRODUÇÃO

1.1 Motivação e Contextualização

O ensino de computação apresenta-se como um desafio diante a rápida mudança no contexto digital e social, os conceitos abstratos unem-se com conteúdos extensivos para garantir a complexidade que é ensinar e aprender computação. Professores que se aventuram na tarefa de ensinar computação necessitam constantes adaptações nos métodos de ensinos ([LAHTINEN KIRSTI ALA-MUTKA, 2005](#)).

Esse desafio pode acarretar no inconveniente e frustrante nivelamento dos alunos em uma turma, de modo que uns compreenderão com certa facilidade certos conceitos e tópicos, outros não terão essa facilidade e em alguns casos podem acarretar em desistência, em baixa produtividade e inclusive em fatores psicológicos. ([EDWARDS, 2004](#)).

Se há o aprendizado ou não, inúmeros fatores influenciam esse resultado. Cytowic, em sua teoria, afirma que o fator emocional possui forte impacto no armazenamento de informações ([BASBAUM, 2002](#)). Partindo desse princípio, se uma plataforma de ensino consegue cativar o aluno de maneira a ser para ele interessante, prazeroso ou até mesmo divertido, a sua eficiência em aprendizado poderá ser maior.

No mesmo sentido, pode-se destacar o processo da aprendizagem significativa, descrito por David Ausubel([MOREIRA, 1999](#)), que propõe que novos conhecimentos tendem a se relacionar com aqueles que os alunos já possuem. Com base nessa consideração de David Ausubel e com o que afirma Cytowic, uma estratégia para o ensino que consiga unir a relação do que é familiar aos alunos com aquilo que os cativa demonstra-se incitador.

Aprendizagem baseada em jogos ou do inglês *Games-Based Learning (GBL)* pode ser definido como uma abordagem de ensino que usa jogos digitais com conteúdo educacional para o auxílio ao aprendizado do aluno jogador([TANG MARTIN HANNEGHAN, 2009](#)). Jogos eletrônicos são ferramentas que tem se tornado cada dia mais familiares, principalmente com o mercado de jogos em crescimento e, constituem elementos que cativam um alto público ([SCHMIDT, 2018](#)). Com isso, aprendizagem baseada em jogos apresenta-se como uma estratégia que liga familiaridade e atratividade, portanto, unindo o que Cytowic e o que Ausubel teorizaram em uma plataforma de ensino.

A aprendizagem baseada em jogos se mostra uma tendência e, usar recursos interativos

faz com que os alunos se atentem ao conteúdo ([SENA SARAH SCHMITHAUSEN SCHMIEGELOW, 2016](#)). Além disso, vale destacar uma das vantagens dessa estratégia citada por ([BORIN, 2007](#)), a de que durante uma partida de um jogo, o jogador se vê forçado a tomar uma postura ativa, vivenciando o jogo, estando em atividade e, se aliando ao ensino da maneira correta.

Assim, a proposta deste trabalho é estudar essa possibilidade também para o ensino superior de computação, produzindo jogos para o aprendizado em Sistemas Operacionais, observando a possibilidade de unir a aprendizagem baseada em jogos ao ensino superior de computação.

1.2 Objetivos

O principal objetivo deste trabalho é desenvolver jogos digitais com o intuito de funcionar como uma ferramenta auxiliar para ensino da disciplina de Sistemas Operacionais para cursos de computação em nível de graduação.

Tais jogos tem sua mecânica e estrutura baseada em conceitos chaves tratados na disciplina, assim, tem-se como premissa a atração do jogador, por seus desafios e diversão, e que consequentemente poderá auxiliar no entendimento do conteúdo educacional, baseando-se na afirmativa de ([PRENSKY, 2012](#)) que será explicada mais a frente neste documento.

Outro objetivo deste trabalho, além de elaborar e desenvolver esses jogos, é estudar a sua eficiência em um estudo de caso, aplicando os jogos em uma disciplina real.

1.3 Organização

No Capítulo 2 é apresentada uma revisão dos conceitos da disciplina de Sistemas Operacionais I. Logo em seguida, cada tópico a ser tratado nos jogos desenvolvidos é apresentado. Também nesse capítulo são apresentadas as ferramentas utilizadas para que fosse possível o desenvolvimento desses jogos.

No capítulo 3 é apresentada a estratégia adotada para que o aprendizado fosse possível nesses jogos. São descritas também a estética dos jogos, tais como escolha do visual e trilha sonora. São ainda descritos cada um dos três jogos desenvolvidos.

Finaliza-se o capítulo 3 discutindo os resultados obtidos por meio da apresentação desses jogos para alunos que cursavam a disciplina de Sistemas Operacionais I no primeiro semestre de 2019 em uma turma do curso de Engenharia de Computação da Universidade de São Paulo no Campus de São Carlos.

Por fim, no capítulo 4 são registradas as contribuições obtidas, as considerações do cenário que o curso apresentou ao projeto e o que se planejou para futuros trabalhos com o que foi observado frente as conclusões.

Capítulo 2

MÉTODOS, TÉCNICAS E TECNOLOGIAS UTILIZADAS

2.1 Considerações iniciais

Os jogos desenvolvidos neste projeto são voltados ao ensino da disciplina de Sistemas Operacionais I. A escolha de manter o foco em uma disciplina foi feita para que a eficiência dos jogos como ferramenta de ensino fosse mais facilmente testável.

Antes do desenvolvimento dos jogos, os conceitos chaves sobre os quais eles se basearam foram estudados e revisados, a fim de que os jogos contemplassem os conceitos principais e então fosse possível auxiliar o aluno no entendimento da disciplina. Esses conceitos são apresentados neste capítulo.

A ferramenta escolhida para o desenvolvimento dos jogos foi a *Unity*, um motor gráfico disponível gratuitamente para Windows e Mac. Todas as trilhas sonoras usadas no jogo foram compostas usando a ferramenta Linux MultiMedia Studio (LMMS), disponibilizada gratuitamente para Linux, Windows e Mac, e todas as imagens foram feitas usando a ferramenta GNU Image Manipulation Program (GIMP).

2.2 A disciplina de Sistemas Operacionais

A disciplina Sistemas Operacionais I constitui uma disciplina obrigatória na maior parte dos cursos de graduação em Computação, fazendo parte do currículo de referência da SBC (Sociedade Brasileira de Computação¹), bem como da IEEE (*The Institute of Electrical and Electronics Engineers*²) e da ACM (*Association for Computing Machinery*³). Sendo assim, constitui uma disciplina considerada de grande importância e que deveria ser aprendida por todos os estudantes da área de computação.

A disciplina apresenta vários tópicos, e em geral é apresentada de maneira teórica aos alunos. Por ser uma disciplina densa e de importância, alguns conceitos chaves foram

¹ <<http://www.sbc.org.br/documentos-da-sbc/category/131-curriculos-de-referencia>>

² <<https://www.computer.org/volunteering/boards-and-committees/professional-educational-activities/curricula>>

³ <<https://www.acm.org/education/curricula-recommendations>>

selecionados, considerando a sua importância, complexidade e também a possibilidade de gamificá-lo, visto em primeira análise, de modo que pudessem ser elaborados em formato de jogos para que a atenção dos alunos fosse obtida.

Os tópicos chaves selecionados para os jogos foram: o funcionamento das *threads* em processamento, os conceitos de semáforos, de região crítica, de gerenciamento de memória, comunicação entre processos, condições de corrida, exclusão mútua, espera ocupada, variáveis de travamento, estrita alternância, primitivas *sleep/wakeup*, escalonamento de processos, estados de processos (indeferido, bloqueado, pronto para execução e em execução), mudanças de estados, algoritmos de escalonamentos preemptivos e não preemptivos ([TANENBAUM, 2010](#)).

Um resumo dos tópicos chaves selecionados é apresentado, de modo que o leitor possa ter uma noção do que se trata em cada um dos conceitos selecionados.

- **Threads**

Threads são definidas como "mini-processos dentro dos processos", ou como partições dos processos em tarefas menores, a serem realizadas praticamente em paralelos.

- **Semáforos**

São variáveis de controle verificadas antes do acesso a uma região crítica. Se a variável possui valor 0, o acesso é negado, se a variável possui valor maior, o acesso é permitido e desconta-se o valor da variável.

- **Regiões críticas**

Também conhecidas como seções críticas, são regiões compartilhadas e disputadas pelos processos em execução cujo acesso simultâneo deverá ser controlado, ou o programa funcionará de maneira irregular.

- **Gerenciamento de memória**

Trata-se de série de estratégias para o uso correto da memória.

- **Condições de corrida**

Refere-se a disputa de regiões críticas. Quando um processo acessa uma região crítica há a necessidade de comunicar essa situação para todos os outros processos.

- **Exclusão mútua**

Diz respeito a ideia de que apenas um processo esteja usando um recurso compartilhado por vez.

- **Espera ocupada**

Também conhecida como espera ociosa, trata-se de estratégias para garantir a exclusão mútua baseada na checagem constante de informações ou na tentativa constante de acessos por parte dos processos.

- **Variáveis de travamento**

Estratégia de exclusão mútua com espera ocupada que se baseia em uso de uma variável com valor igual a 0 quando a região crítica pode ser acessada e diferente de 0 quando a região está ocupada.

- **Estrita alternância**

Chamado de chaveamento obrigatório em (TANENBAUM, 2010, 72), baseia-se na existência de uma variável denominada *turn* que altera de valor, indicando qual processo tem a permissão de acesso.

- **Primitivas Sleep/Wakeup**

Solução para garantir exclusão mútua sem espera ocupada. Um processo realiza a chamada de sistema *sleep* e fica suspenso até que uma chamada de sistema *wakeup* seja realizada.

- **Escalonamento de processos**

Trata-se da tarefa associada ao Sistema Operacional de organizar quais processos serão executados em qual momento e por quanto tempo.

- **Estados de processos**

Um processo pode se encontrar em três estados, bloqueado, pronto e em execução. O processo se encontra no estado bloqueado quando não está preparado para ser executado, pronto quando se encontra preparado ou em execução enquanto estiver sendo processado.

- **Preempção**

Estratégia em um escalonamento que propõe parar a execução de um processo antes de concluir-lo, para executar um outro processo.

Todos esses conceitos dentro de Sistemas operacionais são de grande importância, e poderiam ser descritos em detalhes, entretanto, esse não é o objetivo aqui, de modo que, para um entendimento mais aprofundado dos conceitos o livro Sistemas Operacionais Modernos (TANENBAUM, 2010) pode ser consultado.

2.3 As ferramentas utilizadas

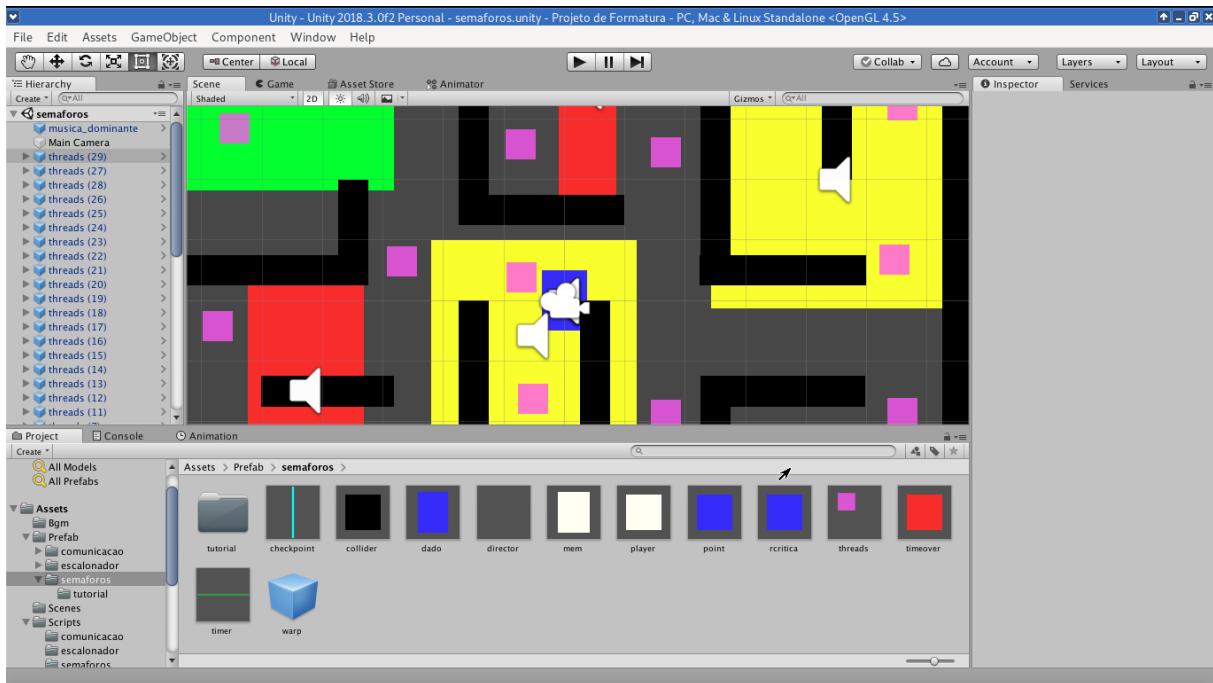
Com objetivo de apresentar os conceitos principais de Sistemas Operacionais de uma forma divertida e cativante, esses conceitos foram apresentados em formato de jogos digitais. Para isso foram utilizadas diversas ferramentas, aqui descritas.

Todas as ferramentas utilizadas estão disponíveis gratuitamente e são conhecidas na área de desenvolvimento de jogos. Esse documento, portanto, atentará a uma breve introdução e a uma justificativa do uso de cada ferramenta.

2.3.1 A ferramenta Unity

A *Unity* é um motor gráfico 3d e 2d muito usado para o desenvolvimento de jogos e se encontra disponível gratuitamente para Windows e Mac ([UNITY, 2019](#)). Um exemplo de uma interface da ferramenta *Unity* no Linux é ilustrado na Figura 1.

Figura 1 – Interface da ferramenta Unity no Linux



A versatilidade em permitir a programação por diversas linguagens e a facilidade da ferramenta em construir jogos para diferentes plataformas foram os principais motivos para a escolha dessa ferramenta no trabalho realizado.

Para desenvolvimento dos jogos nesse trabalho foi usado a linguagem c#.

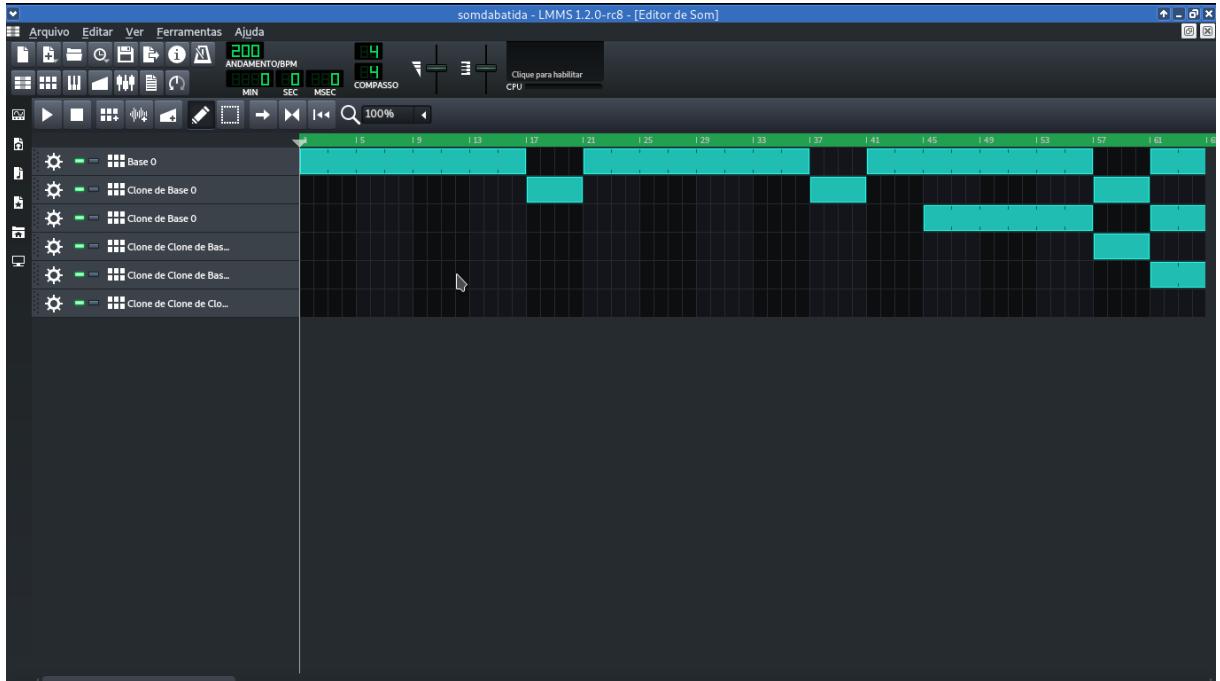
2.3.2 A ferramenta LMMS

LMMS (*Let's make music*) é uma ferramenta para desenvolvimento de músicas simples que permite a construção de arquivos .wav e .ogg. A ferramenta se encontra disponível para Linux, Windows e Mac. e pode ser encontrada facilmente ([NETWORKREDUX, 2019](#)). É uma ferramenta *free*, simples, intuitiva e eficiente para as trilhas sonoras projetadas para este trabalho, por isso essa ferramenta foi escolhida. A Figura 2 ilustra a interface da ferramenta LMMS executando no sistema operacional Linux.

2.3.3 A ferramenta GIMP

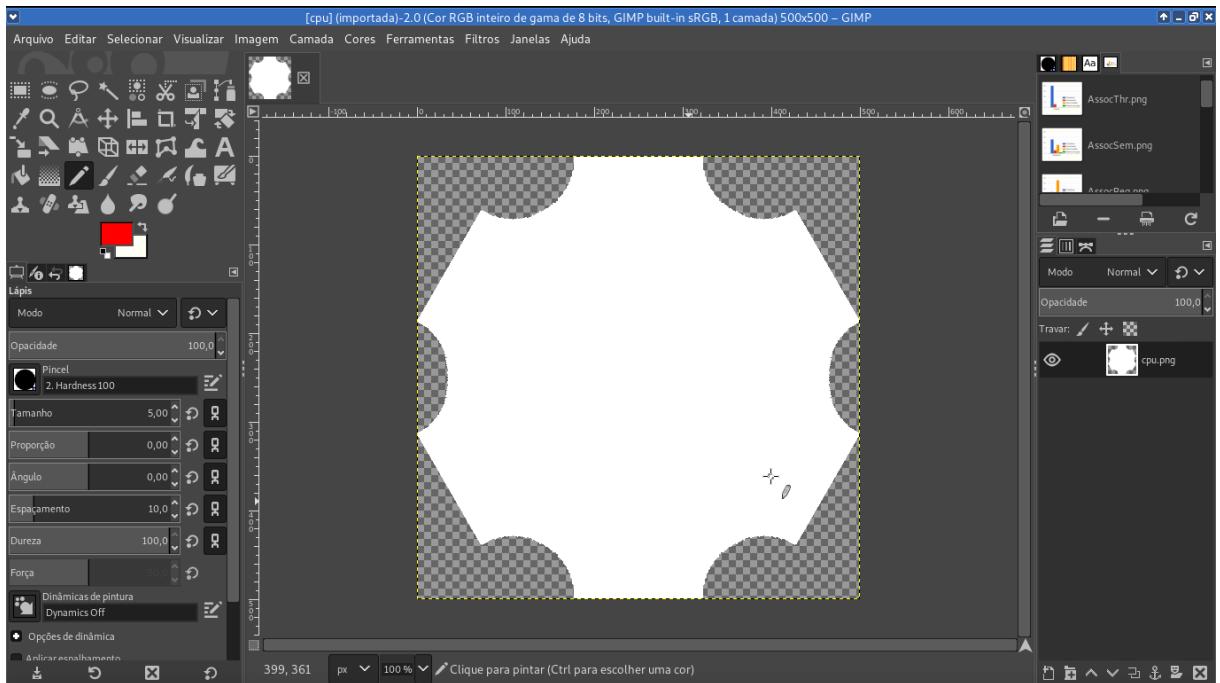
GIMP (*GNU Image Manipulation Program*) é uma aplicação que já se encontra pré-instalada em diversos sistemas operacionais. Trata-se de uma ferramenta de manipulação de imagem simples e prática e está disponível para Linux, Windows e Mac ([GIMP, 2019](#)). A

Figura 2 – Interface da ferramenta LMMS no Linux



ferramenta apresenta suficientes recursos para as figuras geradas para este trabalho. Na Figura 3 é ilustrada a interface da ferramenta GIMP no sistema operacional Linux.

Figura 3 – Interface da ferramenta GIMP no Linux

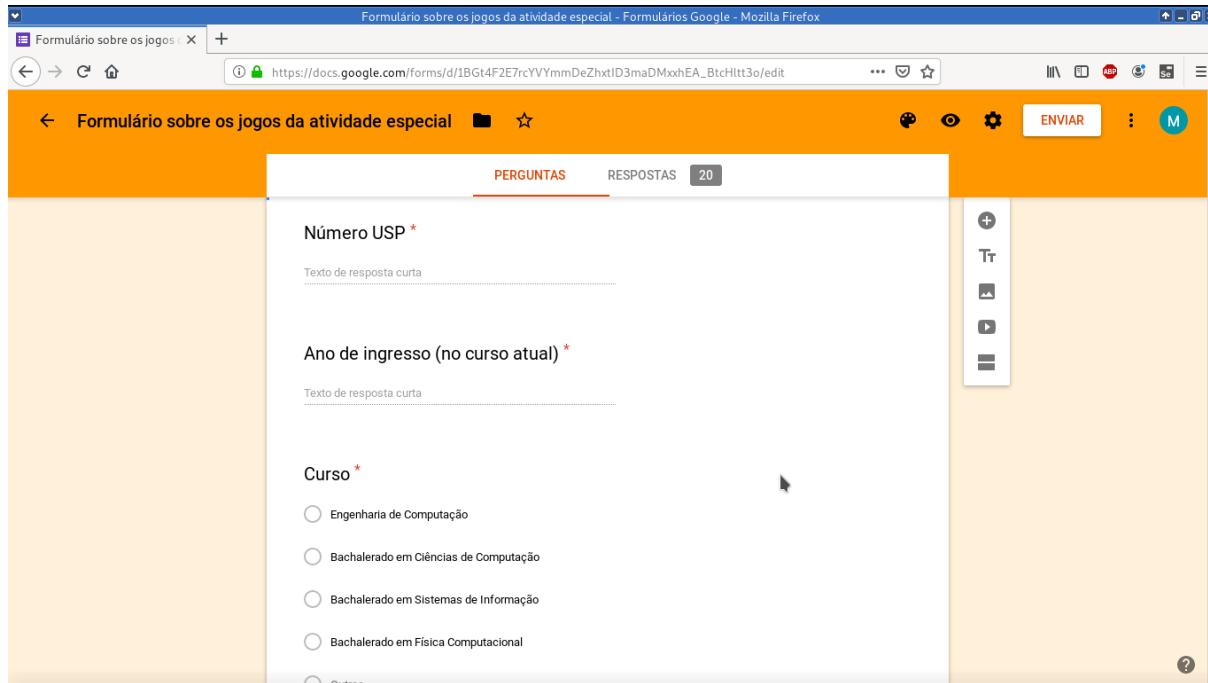


2.3.4 A ferramenta Google Forms

Para que fosse possível a obtenção dos resultados relacionados aos questionários aplicados aos alunos participantes da disciplina de Sistemas Operacionais, foi utilizada a ferramenta

Google Forms. Essa é uma ferramenta *online* e de simples compreensão disponível gratuitamente pela *Google* para criação de formulários e para distribuição do mesmo (GOOGLE, 2019). Na Figura 4 é ilustrada a interface da ferramenta *Google Forms* no navegador Mozilla Firefox executando no sistema operacional Linux.

Figura 4 – Interface do Google Forms no navegador Mozilla Firefox do Linux



Por meio dessa ferramenta é possível se obter tabelas, gráficos e diversas informações úteis sobre as respostas recebidas de maneira prática e versátil. Essas informações possibilitaram a obtenção de resultados interessantes que são tratados na Seção 3.11.

2.4 Considerações finais

Neste capítulo foram apresentadas as ferramentas utilizadas para o desenvolvimento do projeto, que é abordado em detalhes no próximo capítulo. Por se tratar de aprendizagem baseada em jogos, também foi resumido de maneira resumida os diversos tópicos abordados na mecânica dos jogos.

Todas as ferramentas aqui apontadas exigiram um estudo prévio de seus funcionamentos e foram de grande auxílio para o desenvolvimento do projeto.

A revisão da disciplina foi feita de maneira cautelosa para que os tópicos aqui apontados fossem representados de maneira clara, coesa e correta. A preocupação maior se deu com o uso dos conceitos no contexto dos jogos, uma vez que são de grande valia para o aprendizado do jogador.

As estratégias adotadas para realizar essas representações nos jogos são tratadas no

próximo capítulo. É nele também que são explicadas as diversas decisões tomadas no sentido de tornar eficiente a proposta deste projeto.

Capítulo 3

DESENVOLVIMENTO

3.1 Considerações iniciais

Este capítulo tem por objetivo apresentar o trabalho desenvolvido. Foram idealizados e desenvolvidos três jogos, cada um baseado em uma parte do conteúdo diferente da disciplina de Sistemas Operacionais.

Neste documento cada jogo é tratado de maneira particular, mas as escolhas de imagens, som e demais elementos que são similares para os três jogos serão tratados no mesmo tópico, sendo apresentado um para cada elemento.

Assim sendo, na primeira parte, na primeira seção é apresentado o planejamento adotado para o ensino por meio de jogos, qual a ideia da trilha sonora, a proposta do visual adotado, a ideia de implementar um menu dinâmico e as justificativas por trás dos títulos dos jogos e dos tutoriais implementados.

Na segunda parte deste capítulo é tratado cada jogo individualmente, apresentando primeiramente o seu funcionamento, mecânicas e regras, depois associando-o à disciplina e elucidando a estratégia individual do jogo para o ensino da disciplina de Sistemas Operacionais.

3.2 Como aprender com os jogos

Para que os jogos sejam eficientes na tarefa de ensinar, faz-se necessário levar em consideração dois pontos destacados por (PRENSKY, 2012), a mecânica do jogo em si, que deve ser clara, conter desafios e manter uma progressão atraente ao jogador, e ao mesmo tempo o conteúdo do qual o jogo tem a tarefa de ensinar, o dilema nesse ponto é como manter ambos.

A necessidade de diversão do jogo acima do conteúdo educacional também é afirmada por (PRENSKY, 2012), o que não significa em circunstância alguma a desvalorização da segunda, mas a necessidade que a primeira seja efetiva para que a segunda seja absorvida devidamente.

Com isso, a estratégia escolhida ao desenvolver esses jogos incluem manter uma mecânica inspirada na parte conceitual da disciplina que será abordada e, ao mesmo tempo, possibilitar ao jogador tomar estratégias que tem como inspiração algoritmos e soluções reais, também lecionadas na disciplina. Isso significa que o conteúdo educacional está embutido na jogabilidade, no desenrolar de uma partida, presente em tempo real enquanto o jogador está no controle.

Assim, para o jogador alcançar melhores pontuações, será de grande utilidade adquirir conhecimento da disciplina, conhecimento esse, que é apresentado nos jogos. Nesse sentido, ora o jogador aprenderia no jogo a disciplina, ora o jogador aplicaria o aprendido no jogo.

Essa estratégia alia-se com a ideia de liberdade do jogador para planejar e arranjar soluções ao desafio do jogo e assim, descobrir por si mesmo o que o jogo tem a ensinar (MATTAR, 2010), tendo sempre em mente, os pontos destacados em (PRENSKY, 2012).

3.3 Os Jogos

Três foram os jogos desenvolvidos neste projeto: "Threadman", "Race Condition" e Escalonando.

Os nomes do jogos foram formados por trocadilhos e escolha de palavras que remetessem a jogos eletrônicos, assim os títulos tem a pretensão de reafirmar que, além de um auxílio aos estudos, os jogos aqui tratados são formas de entretenimento, como qualquer outro jogo.

A proposta desse formato é que, além de auxiliar o aprendizado do aluno, também traga um ambiente de descontração, pois como destaca (PRENSKY, 2012), essa descontração é necessária e possui um papel no aprendizado.

3.3.1 *Dinamismo do menu e do jogo*

Os três jogos possuem a forma de partidas rápidas, de curta duração, baseado na ideia de que se o jogador errar uma única vez, deverá recomeçar o jogo. A tela de menu dos jogos é ilustrada na Figura 5.

Quando o jogador inicia o jogo, ele se encontra no menu de abertura do jogo e quando o jogador perde uma partida, volta para esse mesmo menu. Como a partida é rápida, isso implica que o jogador normalmente acessará esses menus várias vezes.

Com isso em mente, a construção dos menus tiveram duas prioridades: ser uma tela de transição tão rápida quanto possível e; ser intuitivo e prático.

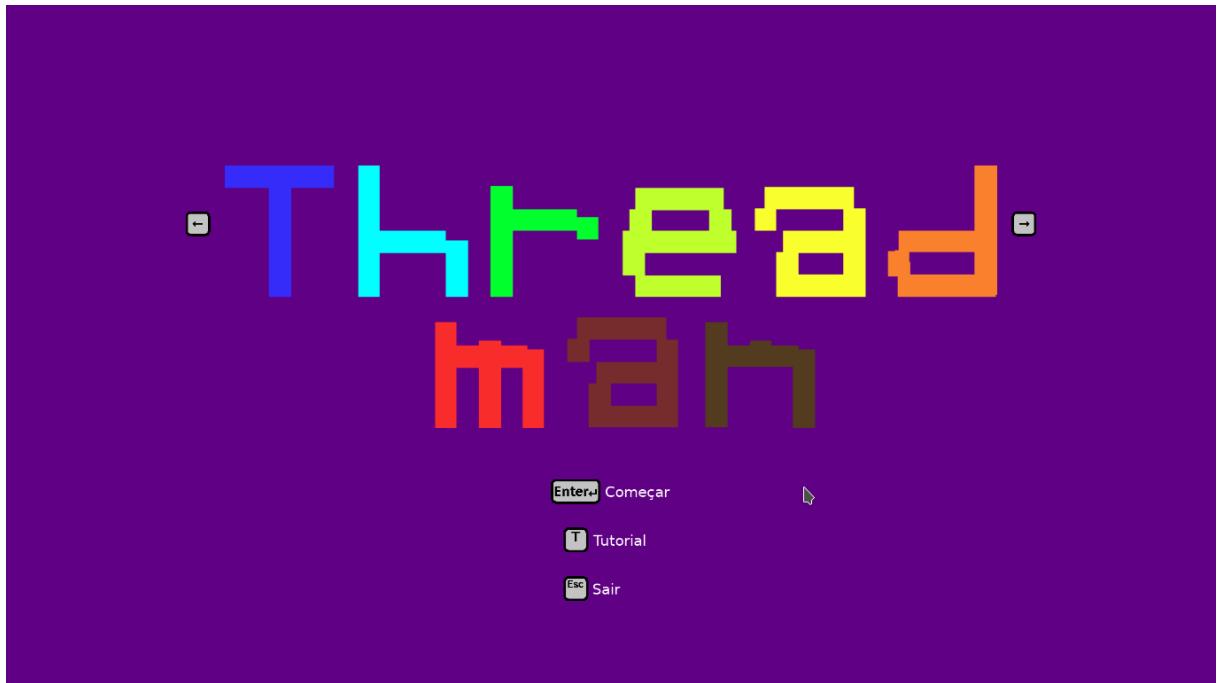
Todos os menus são temáticos e trazem elementos que refletem a mecânica do jogo, além do título do jogo em si. Como os próprios jogos, possuem um visual simplista e fácil de entender.

No menu, o jogador pode apertar a tecla *t* para ler as regras do jogo antes de iniciar uma partida, e em uma partida, também é possível apertar o botão *Esc* do teclado para retornar ao menu quando desejar.

3.3.2 *Os tutoriais*

Para tornar o funcionamento das regras dos jogos claro, tutoriais interativos foram desenvolvidos e disponibilizados na tela do menu dos jogos (Figura 6). Esses tutoriais curtos

Figura 5 – Tela de menu dos jogos.



visam contextualizar a mecânica dos jogos com a disciplina de Sistemas Operacionais, resumindo seu funcionamento e já permitindo a comparação com os tópicos a serem tratados nos jogos.

Figura 6 – Tutorial interativo que proporciona um contexto para o jogador.



Será no momento de jogar, no entanto, que o usuário visualizará e interagirá com esses tópicos, mas os tutoriais já se apresentam como ferramentas importantes para o aprendizado baseado em jogos proposto neste projeto, pois a contextualização está ligada diretamente aos conceitos da disciplina que são elencados nos tutoriais.

3.4 *Threadman, o jogo das threads*

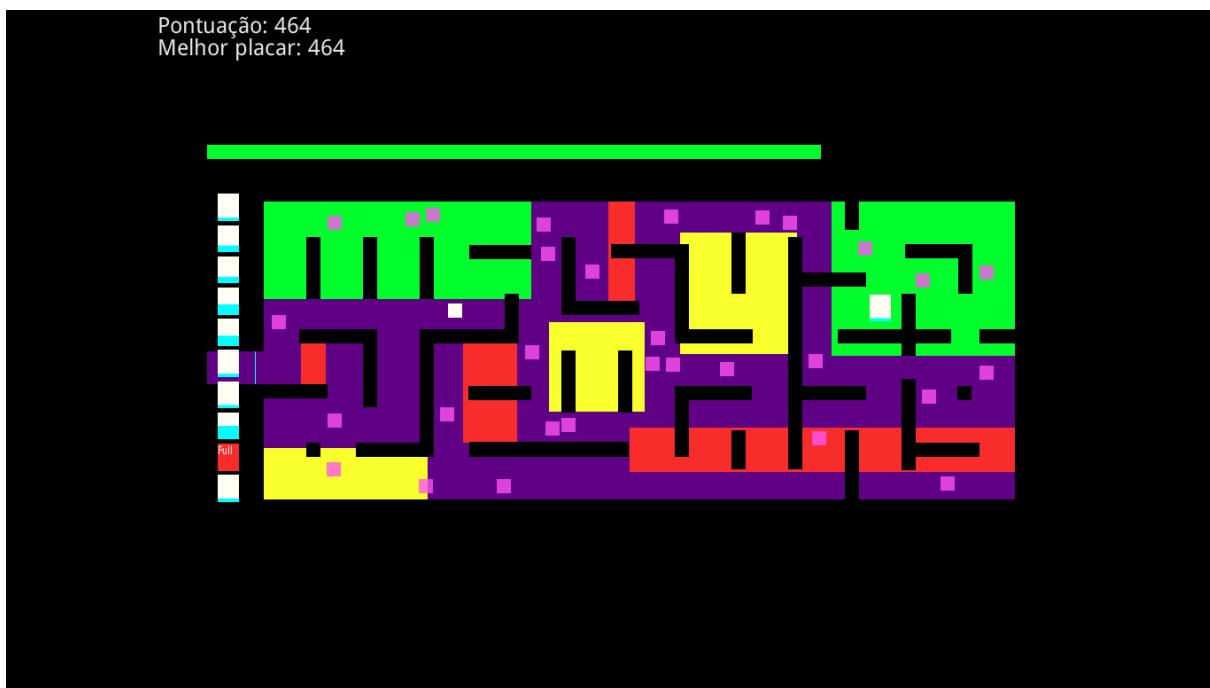
Nesse jogo, são tratados assuntos como o funcionamento das *threads* em processamento, os conceitos de semáforos, de região crítica e de gerenciamento de memória com partições fixas. Além disso, as diversas estratégias de gerenciamento de memória podem ser explicadas com o auxílio desse jogo. O jogador poderá observá-las mudando a sua estratégia de jogo e visualizando as características, vantagens e desvantagens em cada estratégia.

Esse jogo teve inspiração dos jogos clássicos da primeira geração de videogames, tais como o conhecido Pacman e outras plataformas que compartilham este gênero.

Neste projeto foi repensado este estilo de plataforma para a inclusão de regras e elementos estudados na disciplina de Sistemas Operacionais.

Na Figura 7 é ilustrada uma tela do jogo *Threadman*.

Figura 7 – Cena do jogo *Threadman*: respeite os semáforos e resgate a informação.



3.4.1 *Objetivo do jogador*

Nesse jogo é apresentado um mapa com diversos caminhos por onde diversos quadrados rosas percorrem seus trajetos.

Diversas regiões do mapa são preenchidas com as cores verde, amarelo e vermelho, e as demais regiões se mantêm neutras. Na região verde não estarão presentes mais de três dos quadrados, nas amarelas não mais que dois e nas vermelhas no máximo poderá estar presente um quadrado por vez.

O jogador controla um quadrado branco similar aos quadrados rosas, montando o seu trajeto pelo mapa com as setas direcionais. Controlando esse quadrado, o jogador tem como objetivo coletar um bloco de pontos encontrado em algum lugar no mapa (os blocos de pontos são alocados de forma aleatória). Mas para alcançar esse bloco de pontos, o jogador terá que eventualmente passar por uma das regiões coloridas descritas anteriormente.

Para passar nessas regiões, o jogador deverá obedecer as mesmas regras já estabelecidas para os quadrados rosas, isso é, contando com o quadrado do jogador, não pode haver mais que três na região verde, mais que dois na região amarela e apenas um na região vermelha. Caso o jogador desobedeça as regras, a partida é encerrada e o jogo perdido. Nas Figuras 8 e 9 é ilustrada uma cena do jogo *Threadman* com o detalhe da presença de blocos em regiões específicas.

Figura 8 – Cena do jogo *Threadman*: o jogador controla o bloco branco com as setas direcionais. Na região verde, contando com o branco, só poderá haver três blocos.



Após recolher o bloco de pontos, o jogador deve voltar a posição de início, movendo o seu bloco. As regras das regiões coloridas continuam valendo.

Apertando a tecla enter com o quadrado branco na posição inicial, o jogador pode controlar o movimento de uma fileira de retângulos brancos, todos semi-preenchidos com a cor azul. A cor branca de um retângulo representa o quanto de pontos de blocos aquele retângulo ainda suporta e, a parte colorida em azul indica o quanto de pontos de blocos já ocupam aquele retângulo.

Entrando com o bloco branco em um desses retângulos, o jogador irá, enfim, depositar os pontos de blocos que havia recuperado no mapa, pontuando no jogo conforme o tempo que levou para realizar essa tarefa. Cabe salientar que isso só será possível se aquele retângulo ainda suportar esse número de pontos de blocos (ilustrado na Figura 10).

De vez em quando um dos retângulos zeram o total de pontos de blocos carregados, auxiliando o jogador, mas ainda assim ele precisará gerenciar corretamente os seus recursos se

Figura 9 – Cena do jogo *Threadman*: o jogador precisa coletar o bloco de pontos, representado pelo retângulo de faixa azul, mas o bloco branco não pode entrar na região amarela, pois lá já estão presentes dois blocos rosas.

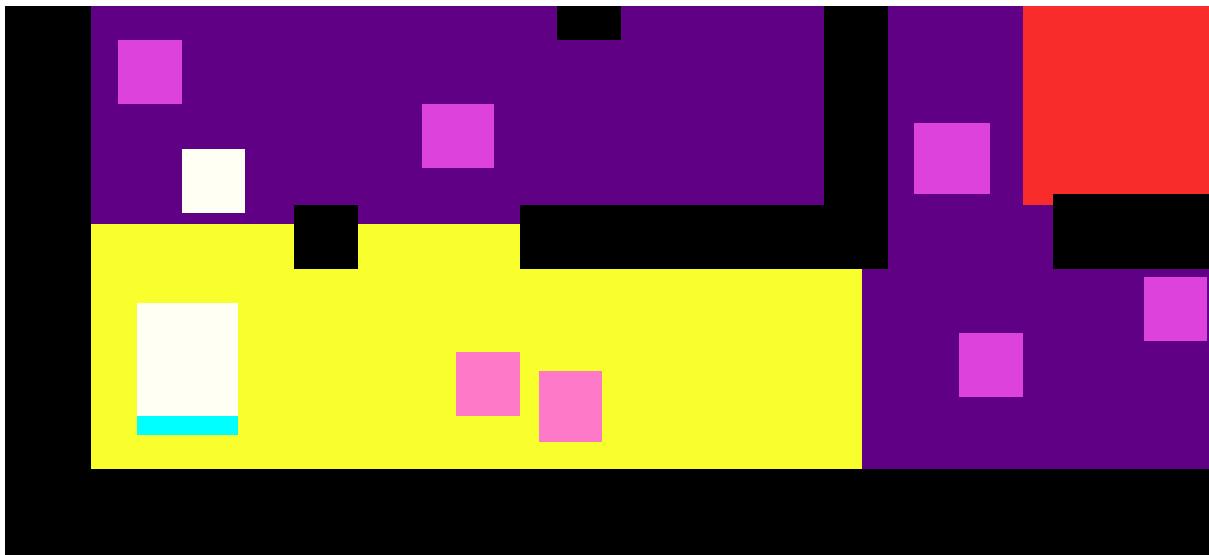
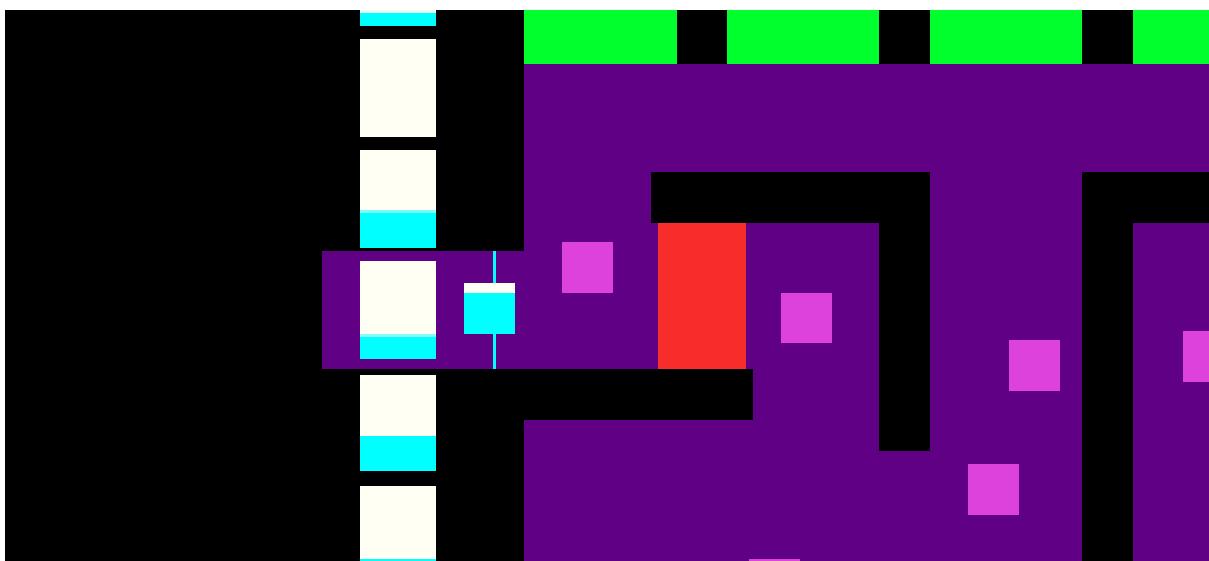


Figura 10 – Cena do jogo *Threadman*: após recolher o bloco de informação e retornar até a posição inicial, o jogador deverá gerenciar o espaço que ele tem disponível nesses retângulos para depositar a informação e pontuar.



desejar ganhar mais pontos.

Toda a tarefa deverá ser realizada dentro de um intervalo de tempo ou a partida se encerrará. A cada vez que o jogador pontua, o intervalo disponível se renova, no entanto sendo menor do que o da última tarefa, tornando o jogo cada vez mais desafiador e incentivando cada vez mais a necessidade de gerenciar os recursos disponíveis.

3.4.2 Relacionando a disciplina com o jogo Threadman

Tanto o bloco branco quanto o cor de rosa representam *threads* funcionando independentes e que compartilham dos mesmos recursos, representado pelo mapa por onde fazem seu trajeto.

As regiões coloridas seriam, nesse contexto, a representação de regiões críticas, com as cores representando a lógica de semáforos, limitando quantas *threads* podem estar ao mesmo tempo compartilhando aquele recurso.

Com exceção do jogador, todas as *threads* obedecerão a regra sempre, e no caso do jogador se recusar a sair de uma região crítica, um exemplo de *deadlock* pode ser demonstrado, pois por causa dele outras *threads* estarão impedidas de usarem aquele recurso e cruzarem aquele caminho do mapa.

Os retângulos na posição inicial do jogo representam um espaço de memória com partições fixas, todas inicialmente com uma parcela já preenchida com dados. Para pontuar no jogo, o jogador deve inserir uma certa quantidade de blocos de pontos nos espaços disponíveis nessas partições, essa quantidade representa dados de memórias a serem armazenados no processo.

Como toda a tarefa deve ser feita em um intervalo de tempo curto, o jogador deverá descobrir estratégias para melhor gerenciar seus recursos, usando o botão espaço para mover os retângulos e conseguir alcançar outras partições. No entanto, o movimento das partições tem velocidade propositalmente lenta, reforçando novamente a necessidade de estratégias de gerenciamento da memória.

Jogando algumas partidas, o jogador poderá descobrir, reforçar e sentir as vantagens e desvantagens de diferentes estratégias como a *first fit*, a *best fit* e a *worst fit* (TANENBAUM, 2010), simulando essas na hora de decidir onde armazenar seus blocos de pontos.

3.5 Escalonando, o jogo do escalonador

Nesse jogo, são tratados assuntos como escalonamento de processos, estados de processos (indeferido, bloqueado, pronto para execução e em execução), mudanças de estados, algoritmos de escalonamentos preemptivos e não preemptivos.

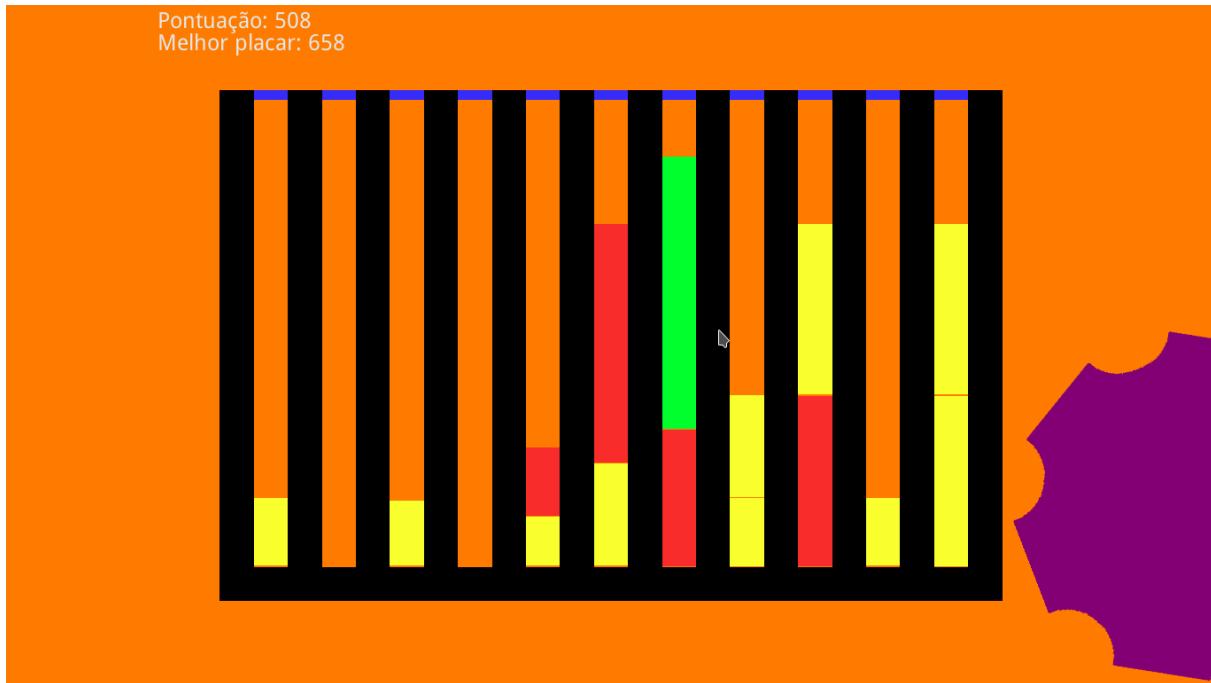
O jogo tem inspiração nos jogos de puzzle clássicos tais como o conhecido Tetris.

Na Figura 11 é ilustrada uma tela do jogo escalonando.

3.5.1 Objetivo do jogador

Ao iniciar uma partida, o jogador se deparará com uma série de setores retangulares distribuídos lateralmente. No topo desses setores existe uma faixa horizontal azul.

Figura 11 – Cena do jogo do Escalonando: escalone os processos e mantenha as filas abaixo da faixa.



Conforme a partida prossegue, retângulos verticais coloridos e de diferentes tamanhos caem sobre esses setores os preenchendo.

Se algum desses retângulo ultrapassar a faixa ao topo dos setores, a faixa muda para a cor vermelha, se essa faixa continuar vermelha por além de um intervalo de tempo de tolerância, a partida será encerrada.

As cores dos retângulos variam entre vermelho, amarelo e verde.

Nada ocorrerá caso o jogador clique com o mouse em um retângulo de cor vermelha, nesse estado não há nada que o jogador poderá fazer, além de esperar. Quando esse retângulo começar a piscar, o jogador poderá enfim tocar nele e fazer com que sua cor mude para amarelo.

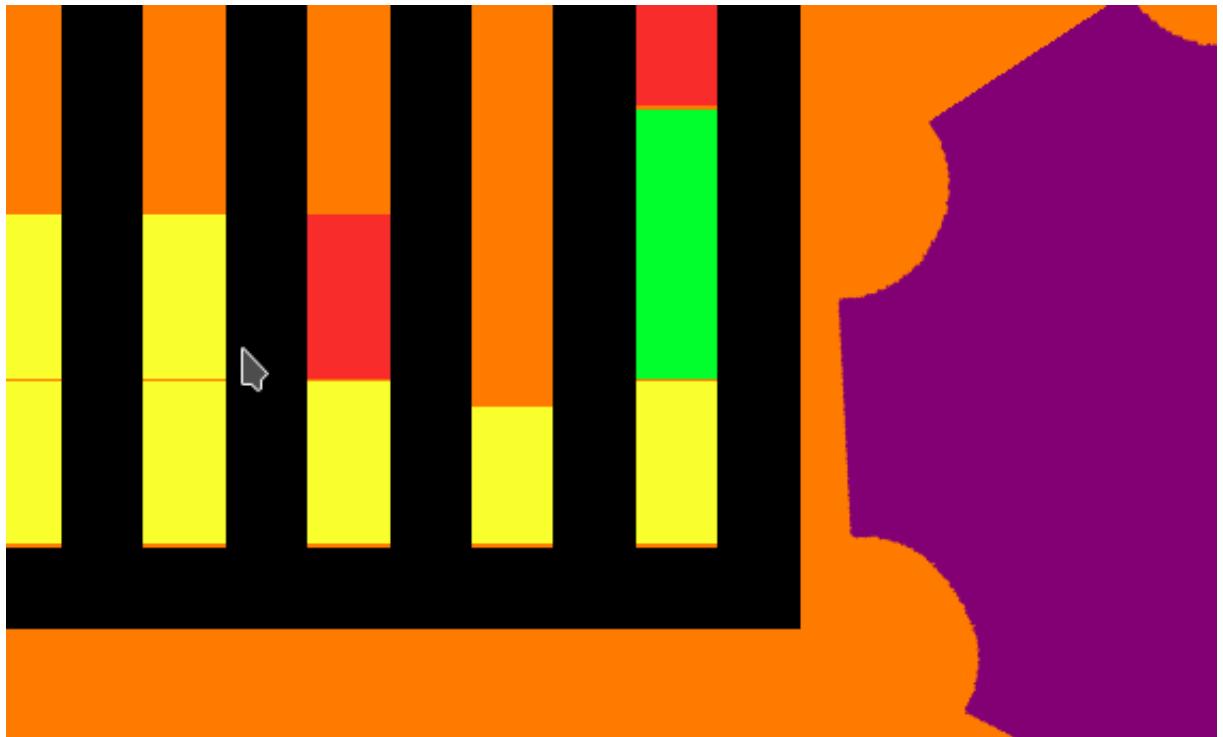
Se o jogador clicar em um retângulo de cor amarela, esse ficará verde e começará a encolher de maneira acelerada até atingir um tamanho mínimo e desaparecer, ao mesmo passo em que uma engrenagem roxa ao fundo começará a girar.

Se o jogador clicar em um retângulo verde, esse irá parar de encolher e ficará amarelo novamente.

Durante a partida sempre haverá apenas um retângulo da cor verde por vez, que será aquele que antes era amarelo e o jogador clicou. Havendo um retângulo verde, não adiantará o jogador clicar em nenhum retângulo amarelo, nesse caso nada acontecerá. Eventualmente, pode ocorrer desse retângulo parar de encolher e também começar a piscar, nesse caso, basta que o jogador clique nele novamente para que sua cor mude de verde para vermelho, assim o jogador poderá clicar em outro retângulo amarelo para fazer com que esse encolha.

O jogador também pode clicar em um retângulo verde mesmo que esse não esteja piscando, nesse caso ele ficará amarelo e deixará de encolher, mas com isso permitirá que o jogador possa escolher outro retângulo amarelo para encolher. Na Figura 12 é ilustrada uma tela o jogo em que são apresentados detalhes do jogo.

Figura 12 – Cena do jogo Escalonando: o jogador tem que montar a sua estratégia para manter as filas abaixo da faixa limitadora.



Resumindo os efeitos dessa mudança, o jogador deverá ficar atento para ver quais retângulos estão disponíveis e pensar em uma estratégia para considerar qual setor necessita ser reduzido, enquanto também deve ter em mente que a qualquer instante um retângulo pode precisar parar de encolher.

O jogador ganha mais pontos por cada retângulo que encolher e os retângulos aparecem na tela com maiores densidades e tamanhos conforme a partida vai avançando, exigindo mais estratégia por parte do jogador.

3.5.2 Relacionando a disciplina com o jogo Escalonando

Os retângulos coloridos representam processos a serem processados por um único núcleo e cada cor dos retângulos representam o estado daquele processo.

A cor vermelha indica que o processo está bloqueado, o jogador não pode colocá-lo em processamento, pois esse se encontra a espera de um evento arbitrário.

A cor amarela representa o processo pronto para execução, ao passar de vermelho para amarelo, significaria que o tal evento arbitrário teria enfim ocorrido.

A cor verde representa o processo em execução, a qualquer momento esse pode acabar por ser bloqueado, por necessidade de mais um evento arbitrário, ou preemptado pelo jogador, por estratégia do mesmo.

Um processo em estado indefinido seria representado por ele não estar sequer no campo de visão do jogador.

As prioridades de cada processo podem ser compreendidas pelo jogador, levando em conta os ganhos em pontos e a regra da faixa limite.

A decisão de quais processos deverão ser executados, em que ordem e da preempção ou não de algum processo cabe ao jogador que faz, portanto, o papel do escalonador.

Nesse sentido, o jogador pode elaborar uma estratégia própria, bem como pode seguir algum dos algoritmos de escalonamentos, tais como o *first-come*, *first-served* ou o *shortest job first*, compreendendo os seus funcionamentos e observando, assim, as vantagens e desvantagens de cada um, também entendendo a importância do escalonador para atender cada processo e diminuir a ociosidade e a fila de processos.

O jogador pode também decidir por jogar sem clicar em um retângulo verde que não esteja piscando, nesse caso estaria simulando um algoritmo não preemptivo. A importância e os problemas com a preempção poderão ser observados com esta estratégia.

Por fim, a intenção da separação dos setores, fazendo com que o jogador tenha que lidar com várias pilhas de processos simultaneamente, é simbólica. O jogo foi planejado dessa maneira para relembrar as funções de um Sistema Operacional em ter que tratar execuções de múltiplos programas, interações de entrada e saída, interações do usuário, armazenamento de memória, entre outras coisas.

Um bom sistema operacional deve conseguir atender todas as tarefas de forma a servir ao usuário da melhor maneira possível, não deixando nada ocioso e sabendo lidar com prioridades, assim como um bom jogador deve conseguir manter todos os setores abaixo da faixa pelo máximo de tempo que conseguir.

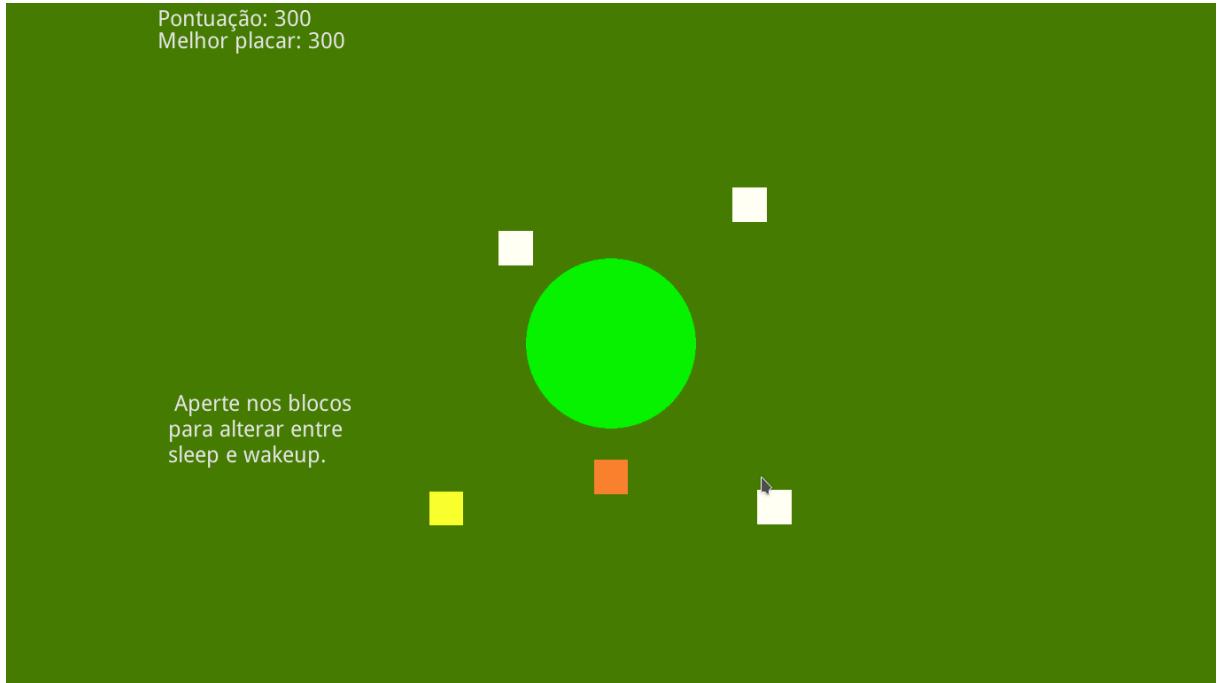
3.6 *Race Condition*, o jogo das comunicações

Nesse jogo, são tratados assuntos como comunicação entre processos, regiões críticas, condições de corrida, exclusão mútua, espera ocupada, variáveis de travamento, estrita alternância e primitivas *sleep/wakeup*.

Esse jogo foi inspirado em jogos populares presentes continuamente nas gerações de jogos, onde a atenção e o ritmo são os principais desafios.

Na Figura 13 é ilustrada uma tela do jogo *Race Condition*.

Figura 13 – Cena do jogo do Race Condition: organize para que todos os processos passem pela região crítica, apenas um por vez.



3.6.1 Objetivo do jogador

O jogo consiste em permitir que apenas um quadrado esteja ocupando a região delimitada por um círculo por vez, no entanto todos os quadrados precisarão passar pelo círculo para concluir o seu caminho.

O jogador contará com diferentes estratégias para interromper o caminho de um quadrado, o fazendo ficar em espera. No entanto, enquanto o quadrado estiver em espera a sua cor irá começar a mudar, de branco para amarelo, depois para laranja, para vermelho, para vinho e, enfim para preto.

Encerra-se a partida se o jogador permitir que dois ou mais quadrados ocupem a região do círculo de uma só vez ou se um dos quadrados chegar a ficar da cor preta. A mudança de cores no jogo é ilustrada na Figura 14.

Figura 14 – Mudança de cores dos blocos que ficam em espera. O jogador não deverá permitir que um bloco chegue a ficar da cor preta, ou a partida se encerra



A cada quadrado que deixa o círculo, o jogador ganha mais pontos, a pontuação ganha por cada quadrado depende da cor que ele apresenta no instante em que deixa o círculo, ganhando mais pontos caso o jogador compreenda como administrar essa situação.

Conforme o jogo avança, aumenta também gradativamente a dificuldade, com maior

densidade de quadrados que desejam acessar a região do círculo e com mais pressa, forçando o jogador a lidar com prioridades.

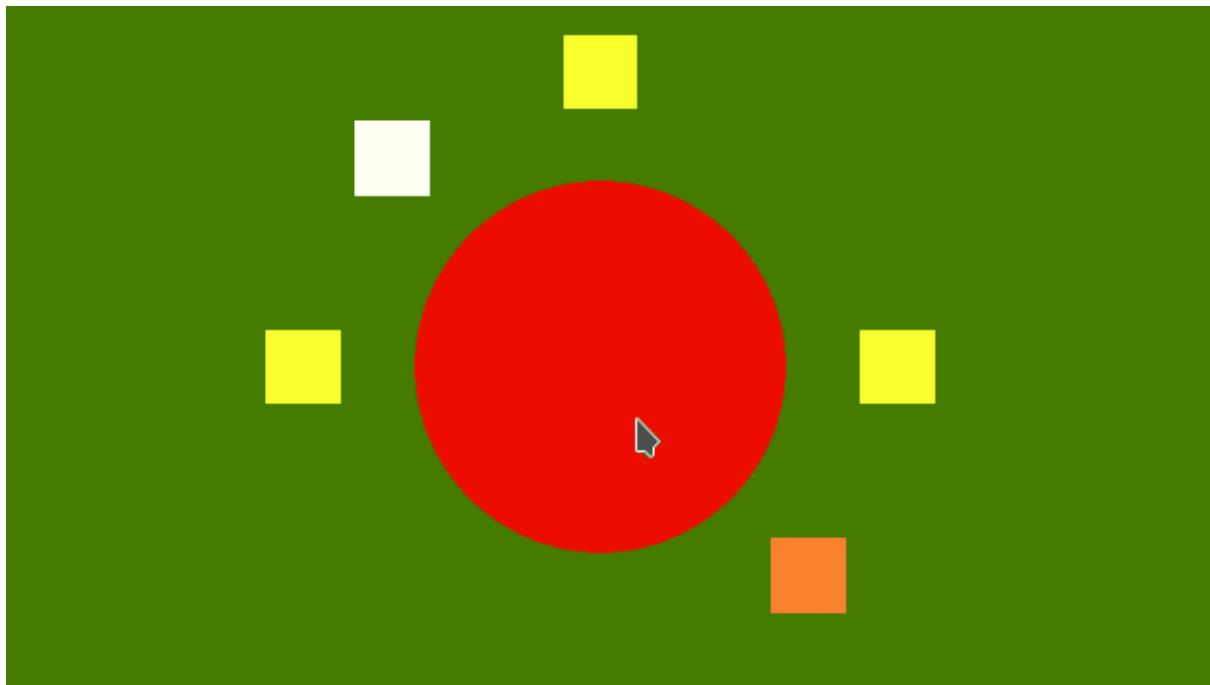
Portanto, o jogador deve priorizar qual a ordem dos quadrados a atravessar o círculo por vez para não perder o jogo e para conseguir alcançar maiores pontuações.

As estratégias para interromper os quadrados são decididas no menu desse jogo e são três, variáveis *lock*, *sleep/wakeup* e *strict alternation*, elas mudam o modo de jogo e para escolher uma delas, basta que o jogador clique em um dos círculos do menu.

- **Modo variáveis *lock***

Nesse modo, o jogador pode clicar com o mouse no círculo, mudando a cor dele de verde para vermelho. Enquanto o círculo estiver vermelho, é garantido que nenhum quadrado entrará no círculo. Na Figura 15 é ilustrada uma tela de jogo em que nenhum bloco entra na região enquanto a mesma encontra-se na cor vermelha, usando a primitiva *lock*.

Figura 15 – Cena do jogo *Race Condition*: no modo variáveis *lock*, nenhum bloco entrará na região enquanto ela estiver vermelha.

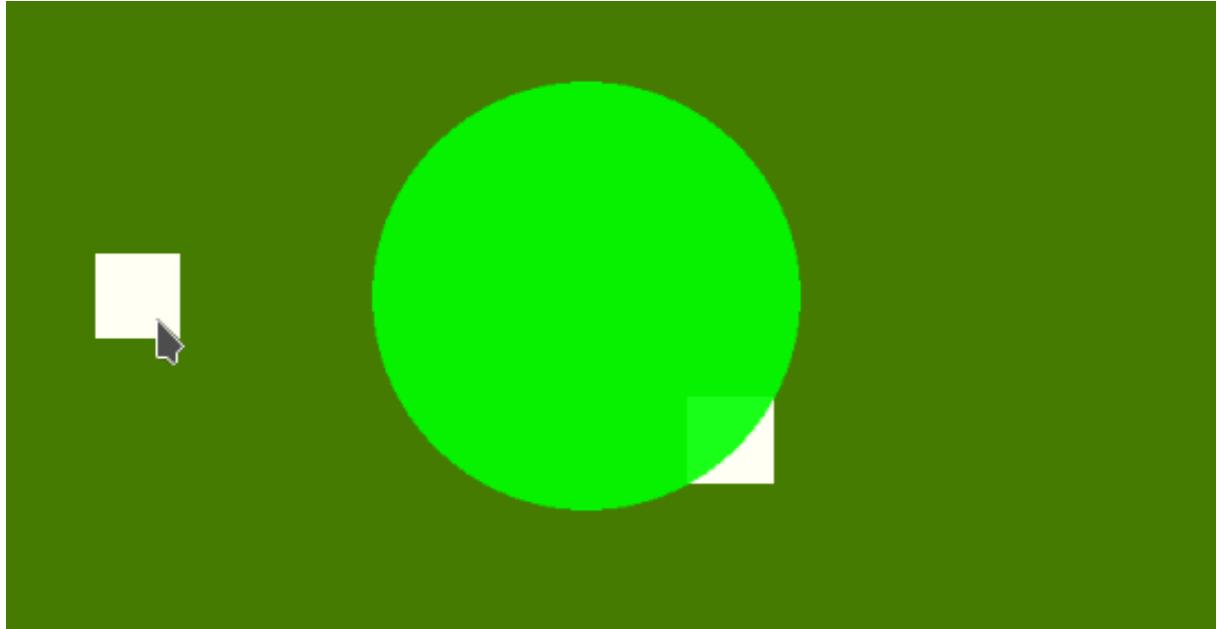


Soltando o botão do mouse, o círculo volta a ficar verde e permite a entrada de qualquer quadrado, mesmo que já haja um dentro do círculo.

- **Modo *sleep/wakeup***

Nesse modo, o jogador pode clicar nos quadrados, fazendo-os pararem de se mover. Um segundo clique nesse mesmo quadrado fará com que ele retorne ao movimento. Na Figura 16 é ilustrada uma tela do jogo *Race Condition* em que o movimento é interrompido e reestabelecido, fazendo uso das primitivas *sleep/wakeup*.

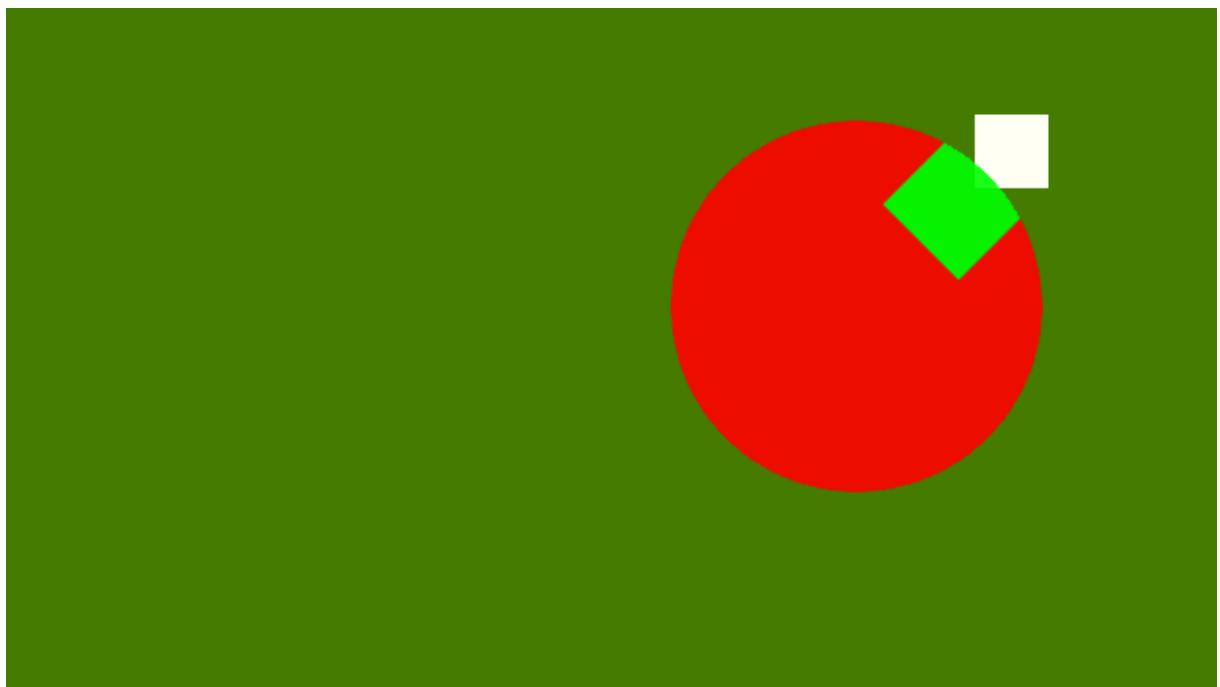
Figura 16 – Cena do jogo *Race Condition*: no modo *sleep/wakeup*, o jogador pode clicar nos blocos para fazê-los pararem e garantir que somente um acesse a região por vez



- **Modo strict alternation**

Nesse modo, o círculo possui um pequeno indicador na ponta, esse indicador refere-se a direção da variável que pode entrar no círculo. Na Figura 17 é ilustrada uma tela do jogo em que pode verificar o uso da primitiva *strict alternation*.

Figura 17 – Cena do jogo Race Condition: no modo strict alternation, o bloco só entra na região se a parte verde estiver virada para ele



As variáveis que não estiverem na direção indicada pelo círculo, não poderão entrar, apenas entrará aquela única que corresponder à direção.

Com um clique no círculo, o jogador poderá girar o círculo em quarenta e cinco graus, mudando a direção permitida para entrada e, portanto, direcionando qual quadrado pode entrar.

3.6.2 Relacionando a disciplina com o jogo *Race Condition*

O círculo representa uma região crítica enquanto os quadrados representam processos que compartilharão recursos dessa região crítica em seu trajeto que representa execuções de tarefas.

O primeiro tópico que o jogador poderá perceber é a lógica da condição de corrida, pois há diversos quadrados, ou processos, vindo em direção ao círculo, ou região crítica, dos quais apenas um deles deverá passar por vez.

Em seguida, o tópico da lógica da espera ocupada pode ser percebido, pois o jogador se vê obrigado a fazer a checagem constante de processos que estão querendo acessar a região crítica, independente do modo escolhido.

Para conseguir pontos, o jogador precisará saber valorizar as prioridades dos processos, representada pela mudança de cor (quanto mais próximo de chegar ao preto, maior a prioridade desse), o que o força a pensar e agir rapidamente. Com isso, as complicações e vantagens de cada método vão se revelando nas partidas, conforme o jogador decide alterá-los no menu.

No modo de variáveis *lock*, a mudança de cor do círculo representa a mudança da variável que indica se a região está livre ou ocupada. O jogador faz o papel da função de verificação, se o processo entrou na região ele deverá clicar no círculo para mudar a variável, se o processo saiu ele deverá clicar novamente e mudar a variável outra vez.

Uma vez que o jogador possui esse papel, ele terá a possibilidade de reconhecer uma das complicações de usar essa estratégia, se ele clicar tarde demais, poderá ocorrer a entrada de mais de um quadrado no círculo, ou de mais de um processo na região crítica. Isso porque não deu tempo de mudar a cor, ou a variável, antes que o segundo quadrado entrasse.

No modo de *sleep/wakeup*, clicar em um quadrado em movimento o fará parar, o que representa o bloqueio do processo, clicando novamente nele, o quadrado começará a se mover outra vez, representando o seu desbloqueio.

A dificuldade nesse método será a de acabar mantendo processos dormindo, ou bloqueados, quando não deveriam estar, o mesmo valendo para processos acordados, ou desbloqueados. Isso ocorrerá por conta da insegurança do jogador para manter a região crítica com a sua capacidade, e representa uma das complicações dessa estratégia.

No método de *strict alternation*, clicar no círculo representa alterar a variável *turn*, que irá decidir qual processo poderá prosseguir, a direção da marca no círculo representa o valor da variável.

Nesse modo, o jogador teria maior facilidade em controlar a entrada para garantir que não haja mais de um processo na região crítica. O desafio nesse modo explora como funciona a mudança de direção do círculo, ao dar um clique, ele gira quarenta e cinco graus no sentido horário, mudando a direção e a permissão de entrada, mas caso o quadrado que estiver livre e não se encontrar na vizinhança daquele que acabou de sair do círculo, poderá ser necessário vários cliques até liberar a sua entrada.

Para aumentar o desafio, a fim de fazer com que o jogador sinta essa dificuldade, o tempo para os quadrados mudarem de cor nesse modo é reduzido. Com isso, há uma reduzida chance do jogador perder por deixar dois quadrados entrarem no círculo, mas há uma chance aumentada dele perder por deixar algum quadrado chegar até a cor preta.

Essa dificuldade representa um dos problemas da *strict alternation*, onde um processo pode estar pronto para usar uma região crítica e a mesma se encontrar desocupada, mas por conta da variável *turn* ainda não indicar a sua permissão, esse continua em espera, quando não precisaria estar.

3.7 Trilha sonora

A importância da trilha sonora em um jogo se dá pela imersão e pelo engajamento do jogador, sendo de grande auxílio para atrair o interesse sobre o jogo ([DIAS FELIPE DE ALMEIDA FREITAS, 2014](#)).

A música em um jogo pode ser observada como responsável por duas funções, a metafórica, ao inserir a atmosfera, a percepção de espaço e tempo, e a metonímica, sustentando a sintaxe do jogo e envolvendo o jogador, o fazendo perseverar ([WHALEN, 2004](#)).

A proposta de cada trilha é incitar o ritmo de cada jogo e desafio.

O *Threadman*, por exemplo, se apresenta como um jogo de estratégia, por isso a sua trilha sonora possui um tom enigmático com um ritmo mais lento e suave.

Enquanto o Escalonando, que é um jogo onde o tempo para agir e as opções disponíveis ao jogador vão se estreitando conforme a partida vai prosseguindo, a música é mais agitada e com a proposta de um ambiente de mais ação.

Já o *Race Condiction*, por se tratar de um jogo onde o elemento ritmo é de crucial importância durante toda partida, a trilha sonora é composta de sons de batuques simples, propondo ao jogador um ambiente de atenção e concentração.

Os menus de cada jogo possuem a trilha composta por remixagem simples da trilha do próprio jogo, a simples ocultação de alguns instrumentos ou alterações de algumas notas ou sequências diferem a trilha do menu com a do jogo, fazendo com que a primeira seja quase como um resumo ou uma introdução da segunda. Cabe salientar que toda a trilha sonora utilizada no trabalho (em todos os três jogos) são de autoria do autor deste trabalho de graduação, consistindo

em músicas originais.

3.8 Visual

O objetivo dos jogos desenvolvidos neste projeto é serem interpretativos. As figuras presentes neles tem representantes abstratos, pois assim são na teoria das disciplinas de computação grande parte das vezes (LAHTINEN KIRSTI ALA-MUTKA, 2005).

A aparência gráfica dos jogos é simplificada. Todos os elementos visuais dos jogos são reduzidos a formas geométricas triviais. Quadrados, retângulos e círculos representam praticamente todos os elementos gráficos nos três jogos e, cores fortes e destacadas diferenciam os elementos presentes na tela. As cores, por diversas vezes, representam estados ou informações e as figuras, funcionalidades simples, buscando manter os conceitos de Human-Computer Interaction (HCI), ainda que em um nível básico (DIX JANET FINLAY, 2004).

A escolha de um visual mais primitivo para os jogos, com formatos simples dos objetos que os compõem, formam um padrão similar aos vivenciados nas gerações um e dois dos video-games. Nessa época, a limitações de hardware obrigavam os desenvolvedores a trabalhar com imagens interpretativas e abstratas, onde as figuras e as cores tinham base em sua funcionalidade simbólica, dentro do contexto do jogo, compreendida apenas quando jogada (LUZ, 2009).

Assim sendo, a escolha do visual para os três jogos foi feita nesses moldes por dois motivos principais: o primeiro deles, é que a complexidade no visual poderia não firmar uma relação de equivalência com a abstração dos assuntos tratados nesses jogos, podendo acabar por ser mal interpretada pelos jogadores (LUZ, 2009), que poderiam acreditar que os jogos são complicados de serem entendidos e, portanto, não cumprirão o papel de ferramentas para auxílio na compreensão da disciplina; O segundo é permitir aos jogadores certo nível interpretativo, para que esses consigam associar os elementos presentes em cada jogo com o conteúdo da matéria, no lugar de causar confusão, caso tal objeto tivesse a forma de algo mais complexo que as formas geométricas simples.

3.9 Estudo de eficiência

Além de apresentar a concepção de jogos para o ensino de Sistemas Operacionais, optou-se também por validar o jogo em termos de eficiência e de interesse para os alunos.

Para poder tirar conclusões sobre o projeto e os métodos, o jogo foi apresentado para alunos de graduação da turma de Sistemas Operacionais I no primeiro semestre do ano de 2019 lecionada pelo professor Dr. Júlio Cezar Estrela que disponibilizou os jogos e um formulário que foi desenvolvido para pesquisa.

Em um total 47 alunos que fazem o curso, 20 alunos responderam ao formulário o que

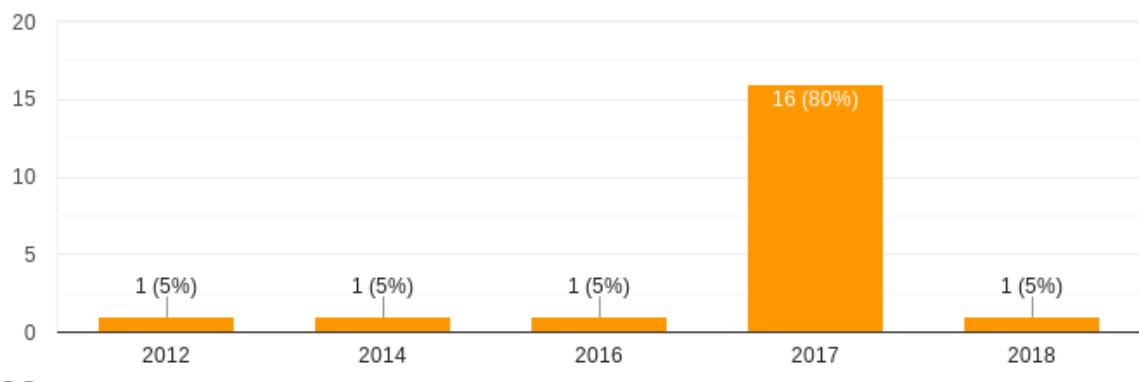
permitiu extrair diversas observações. Desses 20 alunos, 16 ingressaram no curso de Engenharia de Computação em 2017 sendo esses os alunos que se encontram no período ideal para cursar essa disciplina segundo a grade curricular pré-estabelecida, conforme pode ser observado na Figura 18. Cabe destacar que existem alunos de outros cursos, como Bacharelado em Ciência da Computação e Bacharelado em Sistemas de Informação que também fazem o curso.

As questões desse formulário ficaram disponíveis aos alunos para que eles o respondesse da maneira que lhe fosse conveniente, antes, durante ou depois de jogarem. As questões poderiam serem respondidas na ordem que eles desejassesem e mudadas antes da entrega.

Figura 18 – Informações sobre os anos de ingressos e cursos de cada aluno que participou da pesquisa.

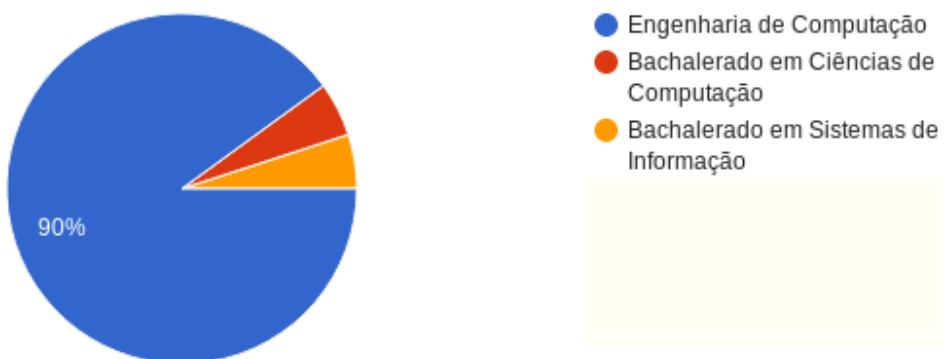
Ano de ingresso (no curso atual)

20 respostas



Curso

20 respostas



3.10 Resultados obtidos

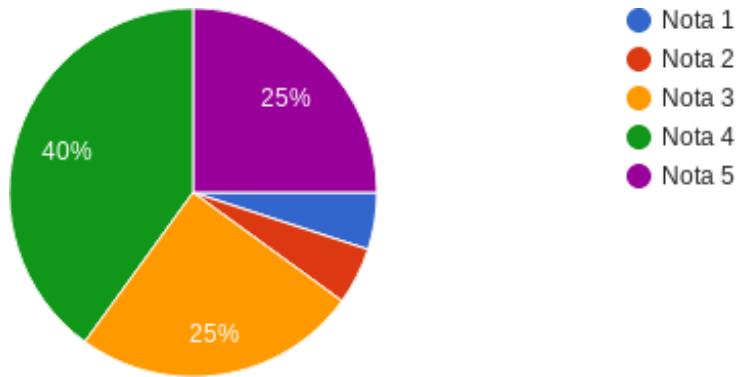
Na primeira parte do formulário foi estabelecida uma série de perguntas de opinião pessoal, em que a resposta dependeria do próprio estudante e de suas experiências e gostos. No entanto foi possível tirar algumas conclusões com relação às respostas obtidas.

3.10.1 Pesquisa de opinião

Em todas as questões dessa primeira parte, foi solicitado aos alunos que estabelecessem uma nota de 1 à 5 para cada item, sendo 1 para péssimo e 5 para ótimo.

Questão 1: Você considera os jogos apresentados nessas atividades divertidos?

Figura 19 – Opinião dos alunos em torno da questão 1.

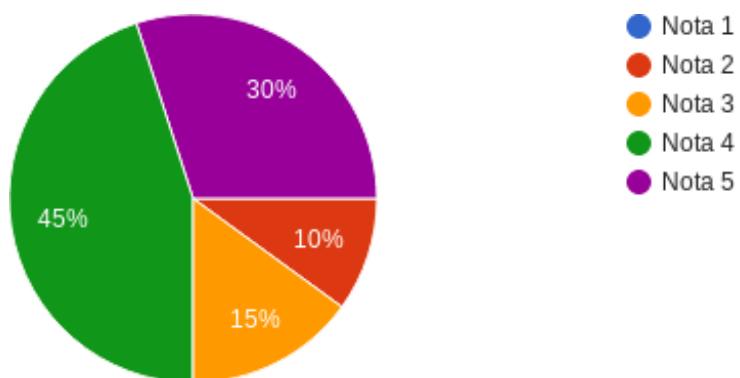


Como já explicado anteriormente nesse documento, é de alta importância que os jogos cativem os estudantes para que a aprendizagem seja eficaz. Não seria teoricamente possível obter a aceitação absoluta, mas deve-se buscar ao menos uma alta aceitação.

Um total de 13 alunos deram notas 4 e 5 nessa questão, 5 alunos deram nota 3 e 2 alunos deram notas 1 e 2. Somados, 90% dos alunos deram nota 3 ou acima (como ilustrado na Figura 19). Pode-se inferir que, apesar da amostragem ainda ser pequena, que os jogos são atrativos aos alunos.

Questão 2: Você considera os jogos apresentados nessas atividades intuitivos?

Figura 20 – Opinião dos alunos em torno da questão 2.

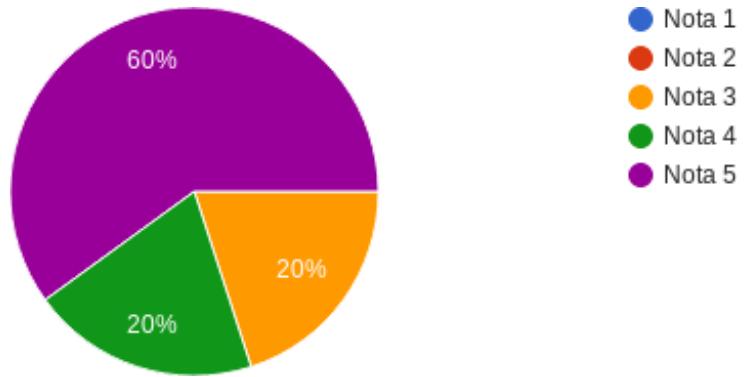


O jogo precisa ser de fácil compreensão para não se tornar inimigo do aprendizado, assim como já mencionado anteriormente nesse documento.

Dos 20, 15 alunos atribuíram as notas 4 e 5, 3 associaram a nota 3 e apenas 2 alunos acreditaram que a mecânica do jogo não era intuitiva (conforme ilustrado na Figura 20). Mais uma vez pode-se observar que os jogos são intuitivos.

Questão 3: Você considera que o conteúdo da disciplina de Sistemas Operacionais presente nesses jogos é comprehensível?

Figura 21 – Opinião dos alunos em torno da questão 3.

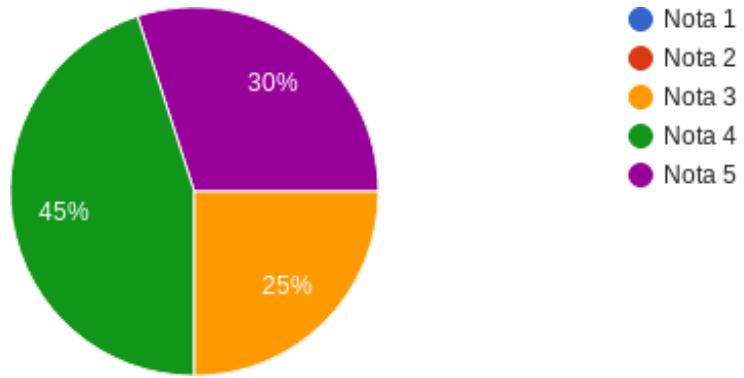


Nessa questão procurou-se avaliar se os alunos julgam ter conseguido enxergar e associar a disciplina à mecânica do jogo e ao ambiente em si.

Não houve nenhuma nota 1 nem 2 na pesquisa, sendo que 12 alunos atribuíram a nota 5, o que permite inferir que os alunos, em geral, acreditaram ter compreendido os elementos da disciplina nos jogos (conforme ilustrado na Figura 21).

Questão 4: O quanto familiarizado você acredita que estava com a disciplina de Sistemas Operacionais antes dessa atividade?

Figura 22 – Opinião dos alunos em torno da questão 4.

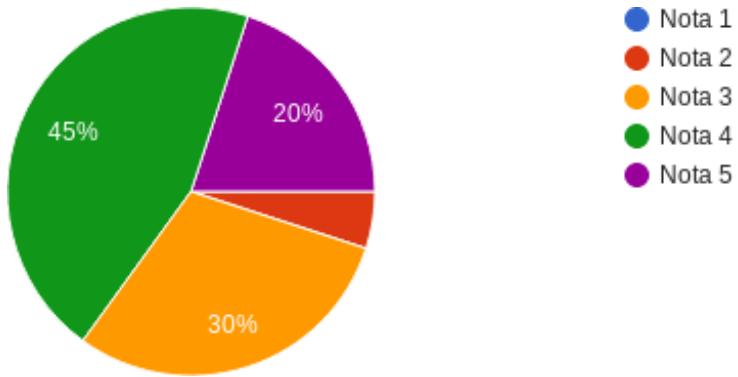


O interesse por trás dessa pergunta é avaliar o conhecimento prévio, ainda considerando apenas a opinião pessoal.

Não havendo nenhuma nota 1 nem 2 e tendo sido apresentado 15 notas 4 ou 5, estabelece-se que os alunos encontravam-se preparados ou assim estavam convictos (conforme ilustrado na Figura 22). Esse tipo de atividade e posterior questionamento por meio de perguntas associadas à disciplina, podem permitir que o docente responsável pela mesma avalie o aprendizado dos alunos.

Questão 5: E para você, pessoalmente, quanto útil ao seu aprendizado na disciplina de Sistemas Operacionais você considera que esses jogos foram?

Figura 23 – Opinião dos alunos em torno da questão 5.

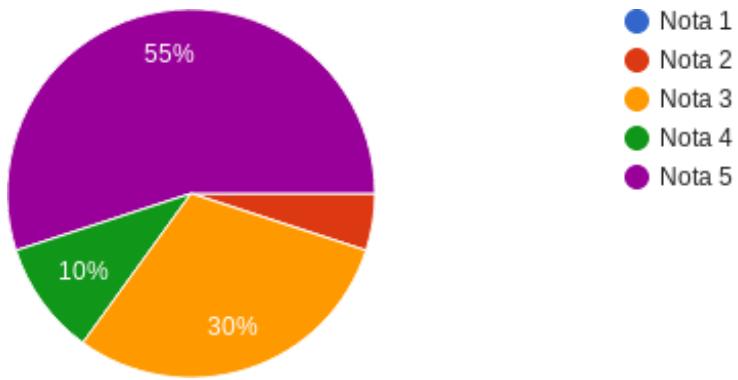


A avaliação do usuário talvez fosse a parte mais importante desse questionário, principalmente considerando que todos os alunos na questão anterior indicaram apresentar alta familiaridade com a disciplina.

Aqui foram associados um total de 13 notas 4 ou 5, 6 notas 3 e uma nota 2. Com isso, somados são 95% (conforme ilustrado na Figura 23 de notas 3 ou maior, o que permite avaliar que os jogos foram úteis para o aprendizado.

Questão 6: Você indicaria esses jogos para os próximos alunos que farão a disciplina?

Figura 24 – Opinião dos alunos em torno da questão 6.



Por fim, foi considerada a opinião dos alunos que testaram os jogos sobre a possibilidade desses jogos serem reapresentados aos próximos alunos, ou seja, se o projeto na opinião deles merecia continuidade.

Somados, 13 alunos deram notas 4 e 5, sendo que deles, 11 deram nota 5. Um total de 6 alunos deu nota 3 e apenas um aluno deu nota 1 (conforme ilustrado na Figura 24). Isso permite inferir que a reapresentação dos jogos em outras edições da disciplina são viáveis e devem ser realizadas. Há ainda a possibilidade de novas avaliações para verificar a efetividade dos jogos.

3.10.2 Perguntas de associações

A segunda parte do questionário possui uma questão mais longa e de associação.

Questão 7: Para cada tópico abaixo, associe os conteúdos da disciplina de Sistemas Operacionais com os respectivos jogos.

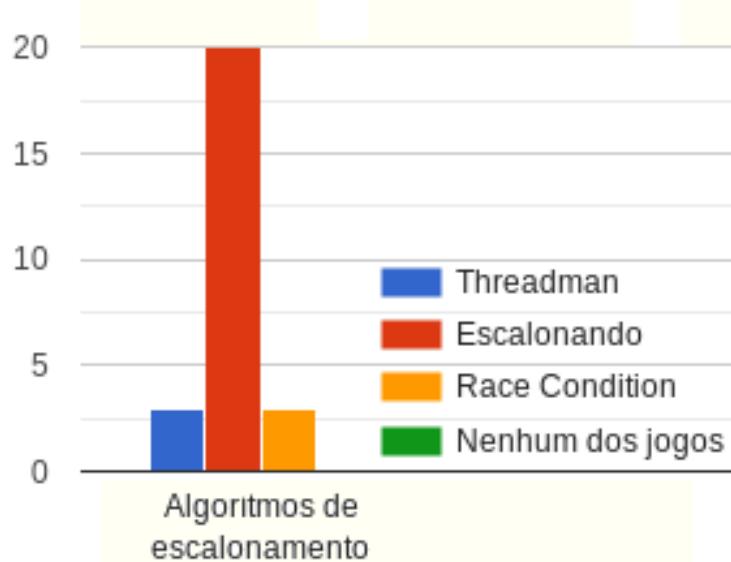
Cabe destacar que nessa questão, os alunos eram livres para assinalarem mais de uma alternativa por vez, podendo haver tópicos a se relacionar com mais de um jogo.

A proposta dessa questão é verificar se os alunos conseguiram associar os tópicos conforme pré-planejados em cada jogo e, assim, confirmar se houve tópicos que deveriam ser melhores explorados.

Por meio das resposta dos alunos, portanto, pode-se obter um ótimo material para aprimorar os jogos, observando em quais tópicos a representabilidade poderia ser aprimorada.

Algoritmos de escalonamento

Figura 25 – Associações feitas pelos alunos dos jogos com o tópico "Algoritmos de escalonamento".



A resposta esperada era o jogo "Escalonando", uma resposta trivial que o próprio título já indica (conforme Figura 25). Os 20 alunos relacionaram esse jogo, mas também tiveram 3 relações com o "Threadman" e 3 com o "Race Condition", o que não era esperado.

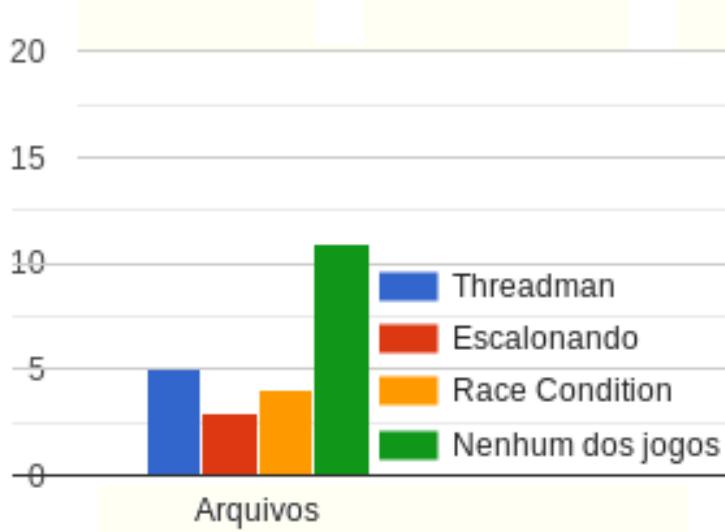
Uma análise levou à conclusão de que essas associações se devem a ideia de escalonar processos para acesso às regiões críticas, mas ainda assim, algoritmos não são tratados nesses dois jogos. A alusão aos demais jogos não é de toda errada, mas os conceitos não são diretamente tratados.

Arquivos

A resposta esperada era "Nenhum dos jogos", 11 alunos assinalaram esse item, o que implica 9 fizeram ao menos uma associação com algum dos outros jogos (conforme ilustrado na Figura 26).

A partir de uma análise verifica-se que possivelmente os alunos não tenham ainda

Figura 26 – Associações feitas pelos alunos dos jogos com o tópico "Arquivos".

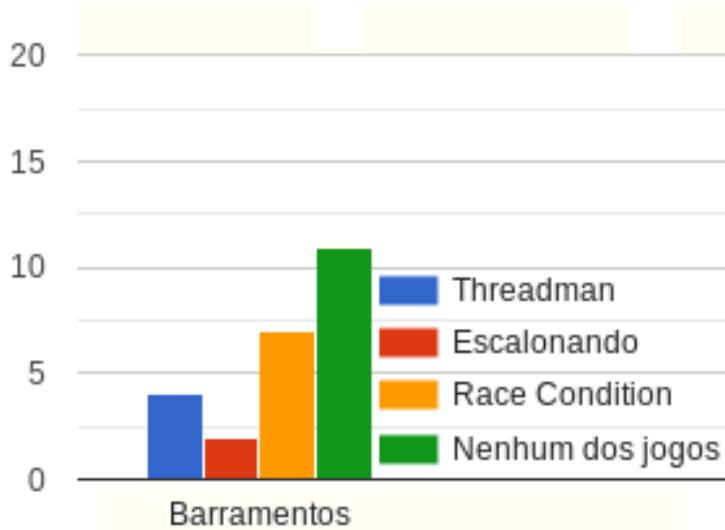


participado de aula sobre arquivos e é possível que tenham associado arquivos com memória ou processos, mas de toda forma foram 5 associações com o "Threadman", 3 com "Escalonando" e 4 com "Race Condition".

Outra possibilidade pode ser o chute, pois não foi colocado no formulário a alternativa de deixar uma resposta em branco, ainda que o esperado nessa questão fosse que quando o aluno não conseguisse associar o tópico com nenhum dos jogos, que ele assinalasse uma opção "nenhuma das alternativas". Um indicativo que apontaria para a possibilidade de chute é a distribuição praticamente uniforme das associações.

Barramento

Figura 27 – Associações feitas pelos alunos dos jogos com o tópico "Barramento".

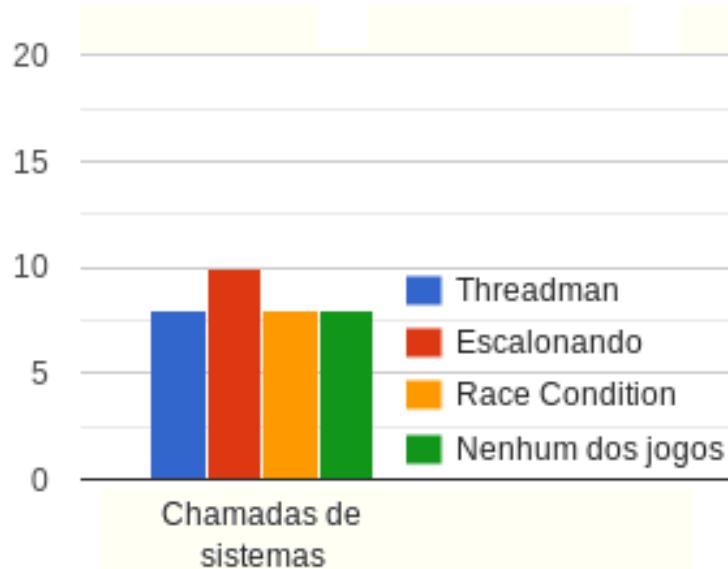


A resposta esperada novamente era "Nenhum dos jogos" e novamente 11 alunos assinalaram esse item (conforme Figura 27).

Houve 4 associações ao "Threadman", 2 ao "Escalonando" e 7 ao "Race Condition". Mais uma vez, nenhuma dessas associações são pertinentes com a proposta dos jogos em si e a uniformidade na distribuição indica a possibilidade de chutes.

Chamadas de sistemas

Figura 28 – Associações feitas pelos alunos dos jogos com o tópico "Chamadas de sistemas".



Essa era outra questão em que se esperava a resposta como "Nenhum dos jogos". 8 alunos assinalaram "Nenhum dos jogos", houve 8 associações tanto para "Threadman" quanto para "Race Condition" e 10 para "Escalonando" (conforme Figura 28).

Aparentemente, há uma resistência por parte dos alunos em assinalar a alternativa "Nenhum dos jogos", uma vez que essa seja a terceira ocorrência. É possível que a falta de cuidado ou a o fato de acreditarem que obrigatoriamente deveria estar associado a um jogo.

Comunicação entre processos

A resposta esperada era "Race Condition", uma vez que somente esse jogo trata do tema de maneira específica.

Houve 14 associações para "Race Condition" (conforme ilustrado na Figura 29), mas as 10 associações para "Threadman" e 6 associações para "Escalonando" são justificáveis, esses jogos não tratam do tópico, mas como tratam de regiões críticas, é possível de se subentender que há comunicações entre os processos.

Condição de corrida

As associações esperadas eram "Threadman" e "Race Condition", sendo que uma associação com o "Escalonando" também seria válida se levado em consideração o algoritmo

Figura 29 – Associações feitas pelos alunos dos jogos com o tópico "Comunicação entre processos".

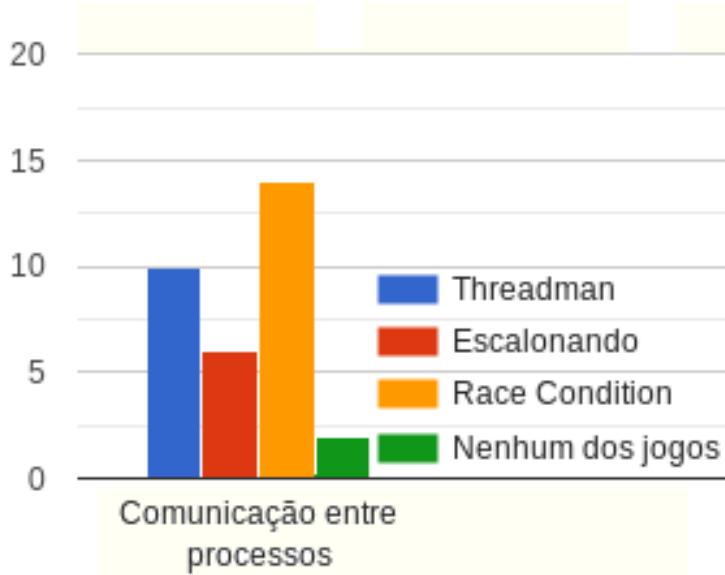
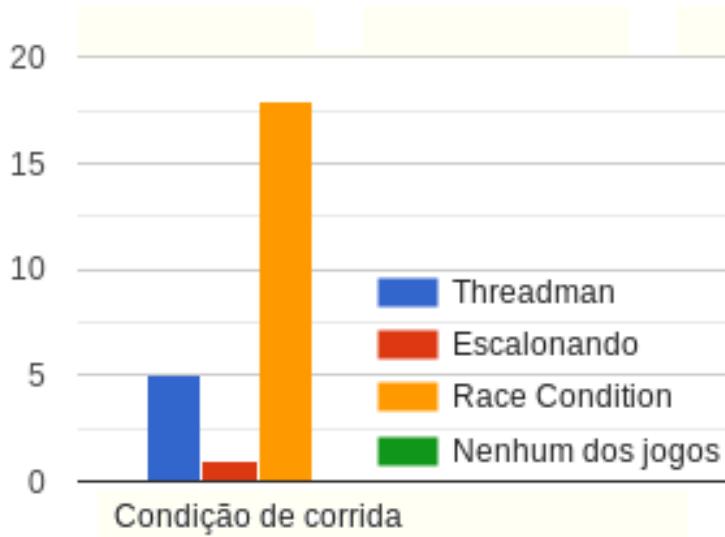


Figura 30 – Associações feitas pelos alunos dos jogos com o tópico "Condição de corrida".



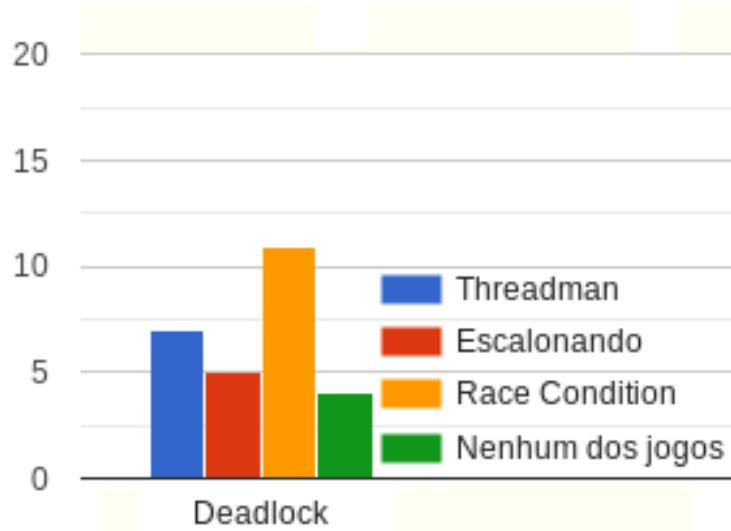
FCFS.

"Threadman" teve 5 associações e, curiosamente, "Race Condition", que leva no título esse tópico, só que em inglês, teve 18 associações, o que significa que 2 alunos não fizeram uma associação trivial nesse formulário. Já o "Escalonando" teve 1 associação apenas e não houve associações a "Nenhum dos jogos" (conforme ilustrado na Figura 30).

Deadlock

Nenhum dos jogos trata do tópico *deadlock* de maneira explícita ou específica, mas em todos eles é possível simular um *deadlock*. Simula-se no "Threadman" ao posicionar a *thread* do jogador em uma região vermelha e não retirá-la de lá, simula-se no "Escalonando" mantendo um

Figura 31 – Associações feitas pelos alunos dos jogos com o tópico "Deadlock".



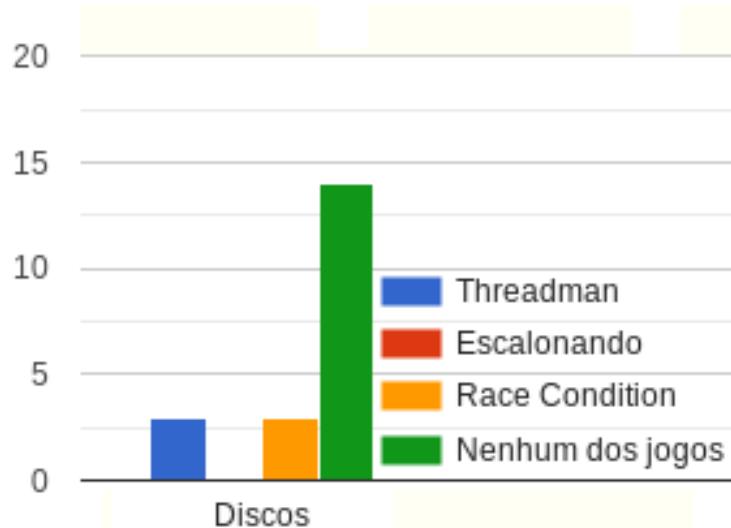
processo travado em execução, simula-se no "Race Condition" clicando no círculo para deixá-lo vermelho e não alterando mais o seu estado.

Assim sendo, não havia uma associação direta ou esperada nesse tópico, mas quaisquer associações seriam pertinentes.

Houve 7 associações para o "Threadman", 5 para o "Escalonando", 11 para o "Race Condition", sendo esse o mais simples de visualizar, o que explica o alto número, e 4 associações para "Nenhum dos jogos"(conforme Figura 31).

Discos

Figura 32 – Associações feitas pelos alunos dos jogos com o tópico "Discos"



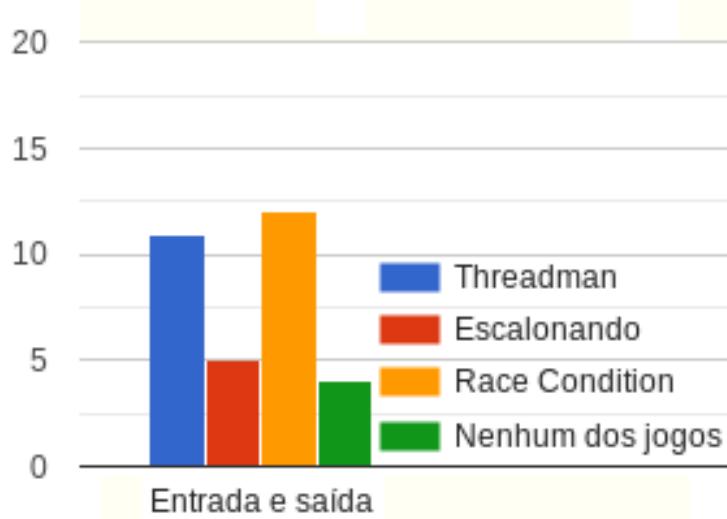
A associação esperada era "Nenhum dos jogos" e 14 alunos fizeram essa associação

(conforme ilustrado na Figura 32).

3 alunos associaram o "Threadman" e 3 associaram o "Race Condition", não sendo respostas pertinentes, pois discos não possuem relações claras com a mecânica desses jogos.

Entrada e saída

Figura 33 – Associações feitas pelos alunos dos jogos com o tópico "Entrada e saída".



A associação esperada era "Nenhum dos jogos", essa foi a resposta de 4 dos alunos (conforme ilustrado na Figura 33).

Ocorreram 11 associações ao "Threadman", 5 ao "Escalonando" e 12 ao "Race Condition". Não foi possível identificar o porque houve a associação desses conteúdos aos jogos. Nenhum deles faz uso desse artifício. É possível que os alunos tenham imaginado o funcionamento interno do jogo, então feito a associação. Entretanto, estudos mais aprofundados fazem-se necessários para que a resposta e a conclusão correta seja obtida.

Escalonamento de processos

A associação esperada era "Escalonando" e o próprio título já traz essa informação, 19 dos alunos fizeram essa associação, restando 1 que não a fez (conforme ilustrado na Figura 34).

Além disso, 3 associações foram feitas ao "Race Condition", no entanto, ainda que esse jogo trate de decidir qual processo poderia acessar uma região crítica por vez, ele não trata exatamente do escalonamento de processos em si. Entretanto é compreensível a associação indevida.

Espera ocupada

A associação esperada era "Race Condition", mas associações aos outros dois jogos são justificáveis, considerando que cabia ao usuário ter a atenção para organizar qual elemento poderia usar dos recursos limitados por vez.

Figura 34 – Associações feitas pelos alunos dos jogos com o tópico "Escalonamento de processos".

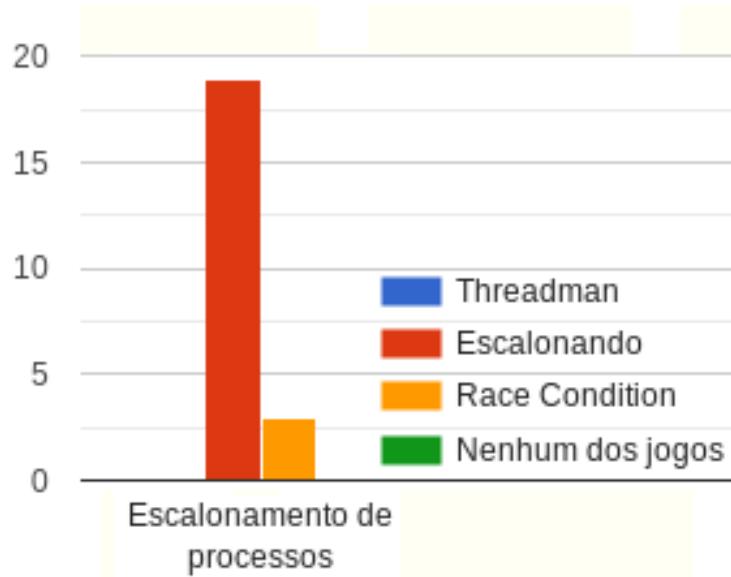
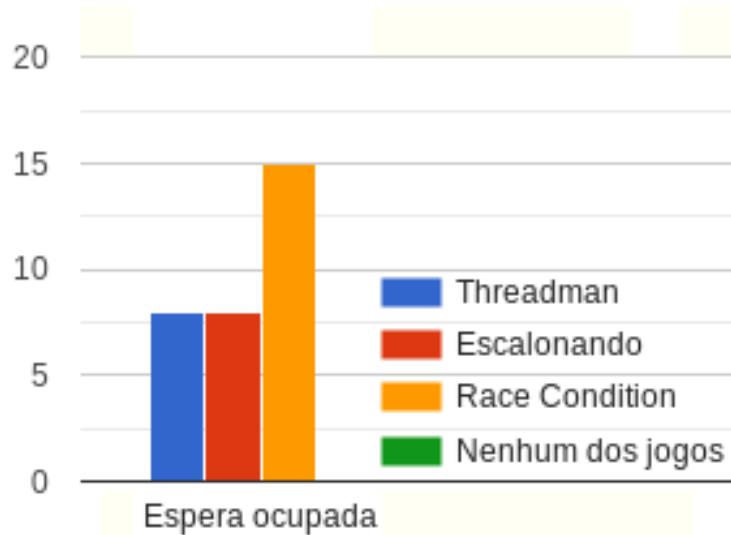


Figura 35 – Associações feitas pelos alunos dos jogos com o tópico "Espera ocupada".



Foram um total de 15 associações ao "Race Condition", o que implica que 5 alunos não fizeram essa associação, sendo que nesse jogo o tópico é um dos elementos centrais (conforme ilustrado na Figura 35).

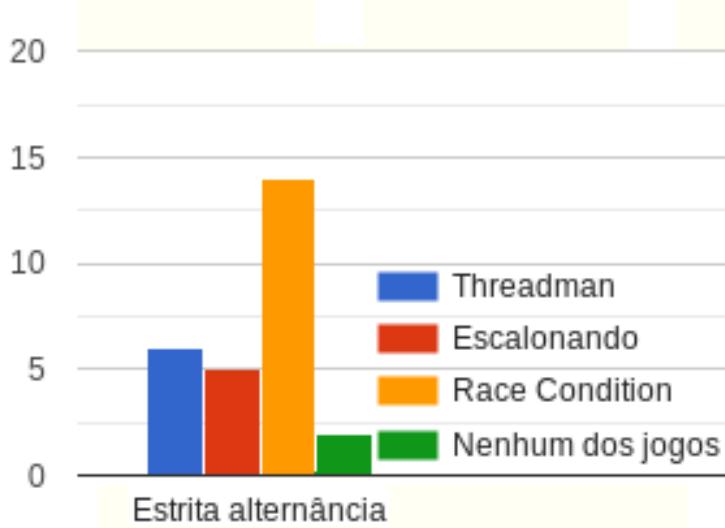
Além disso, houve 8 associações tanto para o "Threadman", quanto para o "Escalonando" e não houve nenhum aluno que assinou "Nenhum dos jogos".

Estrita alternância

A associação esperada era "Race Condition", 14 alunos fizeram essa associação. Dos 6 que não a fizeram 2 assinalaram "Nenhum dos jogos" (conforme ilustrado na Figura 36).

Houve também 6 associações ao "Threadman" e 5 ao "Escalonando", ambas não podem

Figura 36 – Associações feitas pelos alunos dos jogos com o tópico "Estrita alternância".

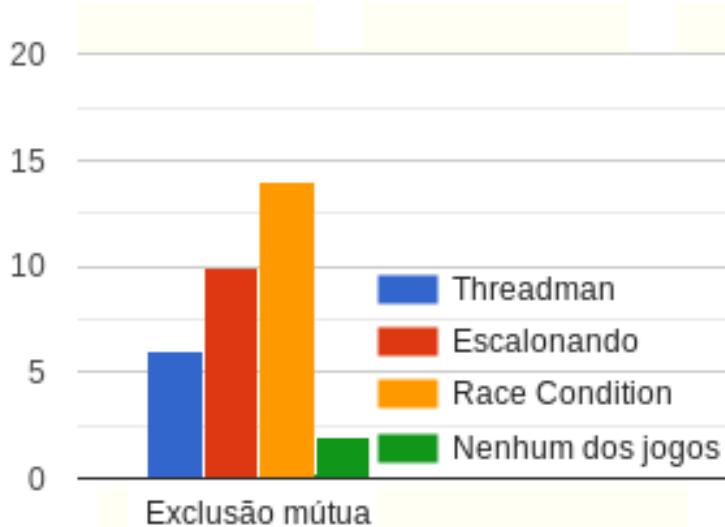


ser consideradas pertinentes, pois nenhum desses jogos tratam desse algoritmo.

Um dos modos do jogo "Race Condition" tem o tópico no título, o que leva a possibilidade de uma falta de atenção por parte dos 6 alunos que não realizaram a associação.

Exclusão mútua

Figura 37 – Associações feitas pelos alunos dos jogos com o tópico "Exclusão mútua".



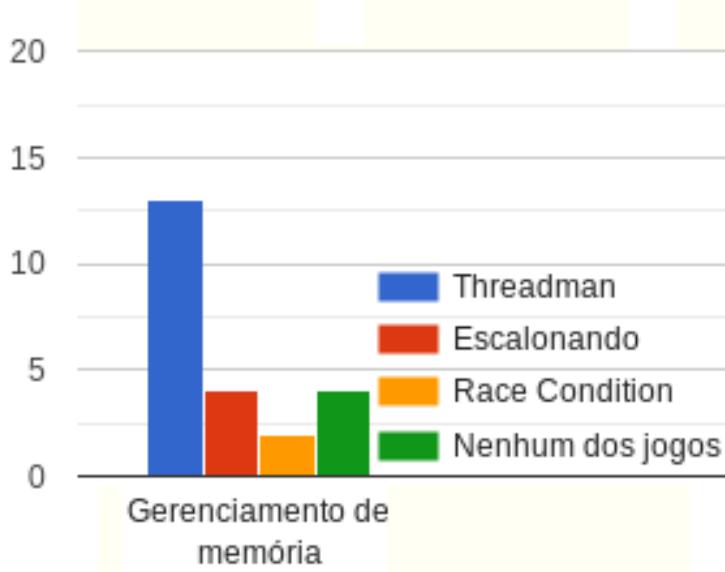
Pode-se associar aos três jogos, a única resposta não esperada era "Nenhum dos jogos" e foi a resposta de apenas 2 alunos (conforme ilustrado na Figura 37).

Foram feitas 6 associações ao jogo "Threadman" onde possivelmente o tópico tinha maior destaque, pois o jogador tinha que obedecer o princípio da exclusão mútua para prosseguir com o jogo.

Houve também 10 associações ao "Escalonando" e 14 ao "Race Condition", com isso se conclui que, apesar do tópico estar embutido nas regras do "Threadman", ele foi mais notado nos outros dois jogos.

Gerenciamento de memória

Figura 38 – Associações feitas pelos alunos dos jogos com o tópico "Gerenciamento de memória".



A única associação esperada era o "Threadman", 13 alunos fizeram essa associação e, portanto, 7 que não a fizeram, sendo que 4 desses assinalaram "Nenhum dos jogos" (conforme ilustrado na Figura 38).

O gerenciamento de memória é uma das estratégias importantes para pontuar melhor no jogo "Threadman", no entanto, se essa estratégia for ignorada, de fato esse tópico pode passar despercebido.

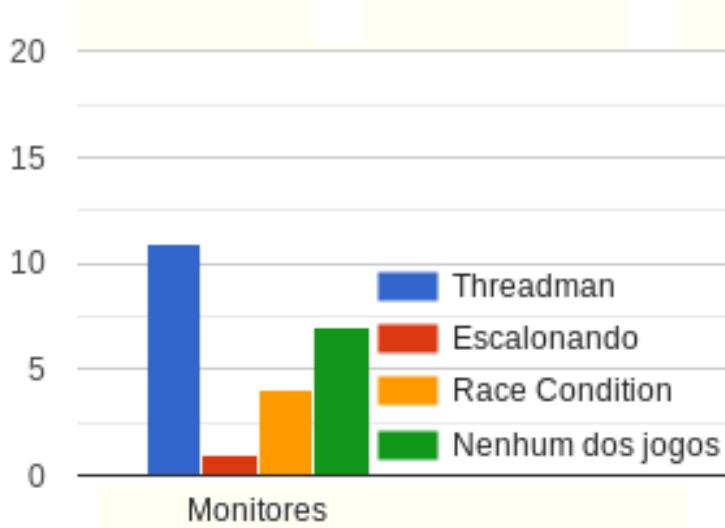
As 4 associações ao "Escalonando" e as 2 ao "Race Condition" não são pertinentes, o tópico não é tratado de nenhuma maneira nesses jogos e essas associações talvez estejam interligados com possíveis chutes, como já mencionado em questões anteriores.

Monitores

A associação esperada era "Nenhum dos jogos". 7 alunos associaram como esperado e 11 alunos acabaram por criar associações com o "Threadman" (conforme Figura 39), no entanto, como explicado em (TANENBAUM, 2010, 82), monitores seriam parte de uma estratégia que impossibilita a violação da região crítica, pois ao tentar acessar a região crítica, o processo espera instruções que verificam se há um processo já na região e bloqueia o processo que deseja entrar se isso for confirmado.

O conteúdo proposto pelo jogo "Threadman", como já explicado, é semáforos e não monitores, os semáforos (TANENBAUM, 2010, 77), por outro lado, seriam uma variável inteira cujo valor indica quantos processos poderiam acessar a região crítica. Quando incorretamente

Figura 39 – Associações feitas pelos alunos dos jogos com o tópico "Monitores".

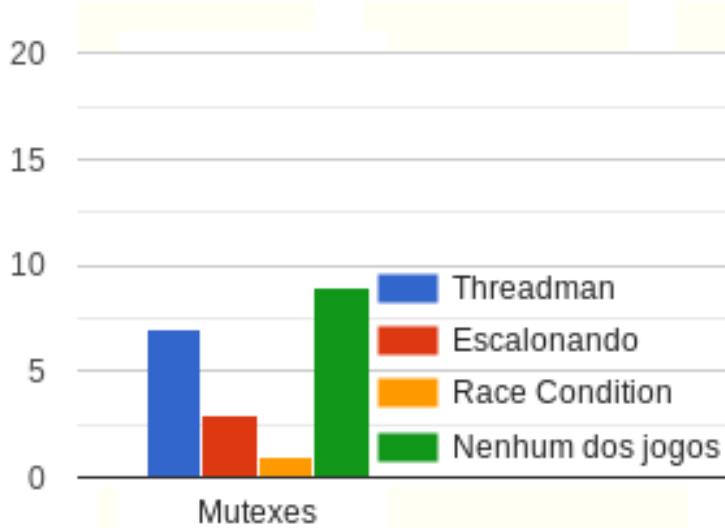


manipulado, semáforos podem acabar por violar a região crítica. Outra diferença a se apontar é que monitores só permitem o acesso de um processo na região crítica e semáforos não necessariamente.

Ocorreu também 1 associação ao "Escalonando" e 4 ao "Race Condition", a única associação ao "Escalonando" não é pertinente e talvez as 4 associações ao "Race Condition" sejam devido ao modo de estrita alternância, onde ao apertar o círculo que representa uma região crítica, ele bloqueia a entrada de qualquer processo, tal como num monitor. Ainda assim, não seria uma associação correta, pois estrita alternância baseia-se em espera ocupada.

Mutexes

Figura 40 – Associações feitas pelos alunos dos jogos com o tópico "Mutexes".



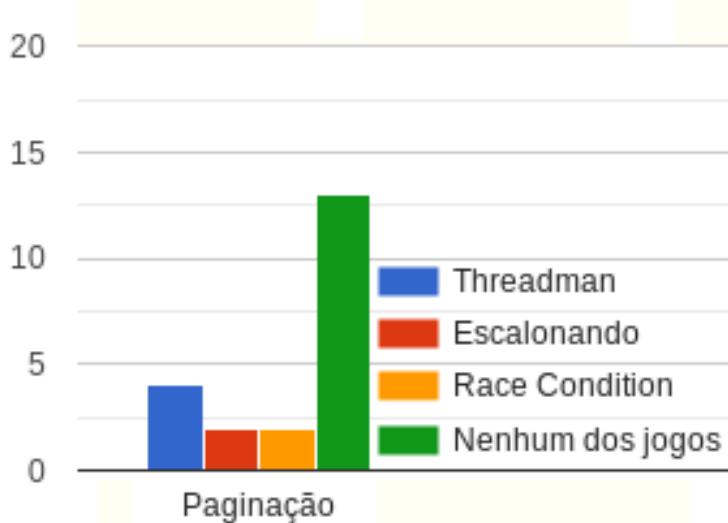
A associação esperada era "Threadman", uma vez que um semáforo binário atue de maneira similar a um mutex (TANENBAUM, 2010, 79).

Foram feitas 7 associações ao jogo "Threadman", 3 ao "Escalonando" e 1 ao "Race Condition", as associações a esses dois últimos jogos não são pertinentes e devem estar associadas a chutes e conclusões errôneas (conforme ilustrado na Figura 40).

Houve também 9 alunos que assinalaram "Nenhum dos jogos", que é uma resposta válida se considerarem a semelhança, pois o tema do jogo é semáforos.

Paginação

Figura 41 – Associações feitas pelos alunos dos jogos com o tópico "Paginação".



A associação esperada era "Nenhum dos jogos" e 13 dos alunos assim associaram (conforme Figura 41).

Foram feitas 4 associações ao "Threadman", 2 ao "Escalonando" e ao "Race Condition", nenhuma delas seria pertinente, não havendo qualquer lógica simples e aparente dos motivos de tais associações a não ser, talvez o chute ou algum mal entendido. Como nos demais casos, experimentos com mais alunos e mais aprofundados são necessários para que as afirmações sobre os jogos possam ser melhor realizadas.

Preempção

A associação esperada era "Escalonando" e 18 alunos fizeram essa associação (conforme ilustrado na Figura 42).

Foram feitas 4 associações ao "Threadman" e 4 ao "Race Condition", em ambos os jogos há momentos que algum processo ou *thread* tem que interromper suas atividades, é provável que os alunos tenham interpretado como uma preempção.

Primitivas sleep/wakeup

Figura 42 – Associações feitas pelos alunos dos jogos com o tópico "Preempção".

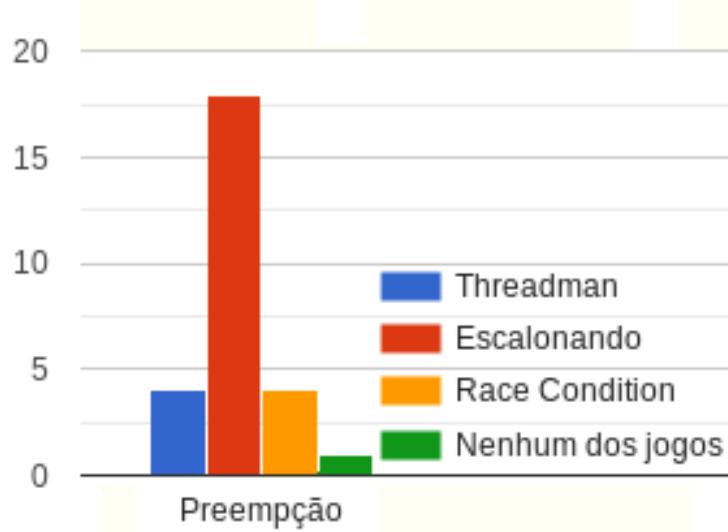
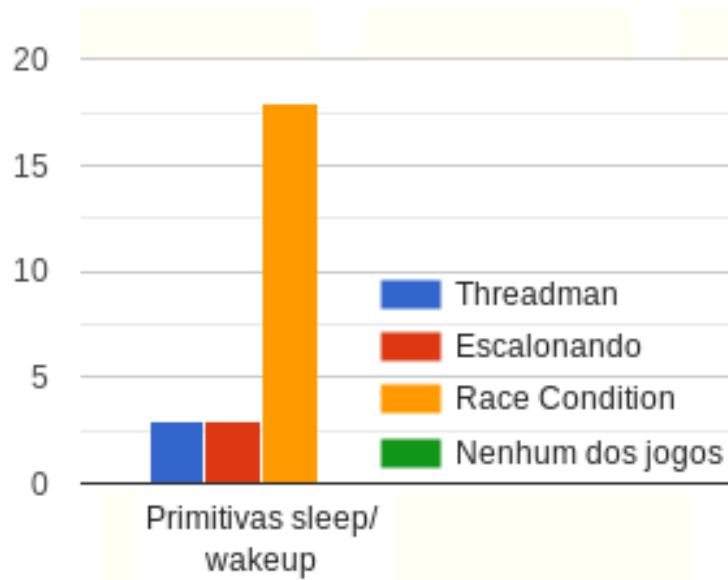


Figura 43 – Associações feitas pelos alunos dos jogos com o tópico "Preempção".

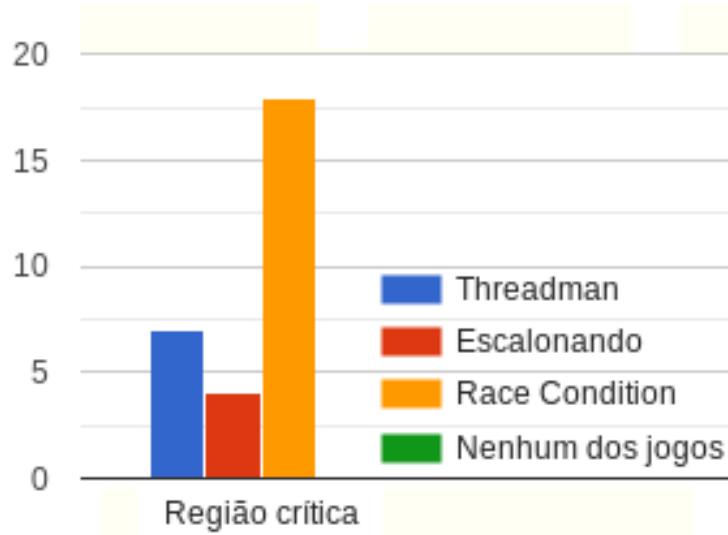


A associação esperada era "Race Condition" e, novamente, 18 alunos fizeram essa associação (conforme ilustrado na Figura 43). O nome de um dos modos desse jogo é *sleep/wakeup*, então não seria lógico a falta dessa resposta por parte de dois alunos.

Foram feitas 3 associações tanto para "Threadman", quanto para "Escalonando". Embora em "Threadman" existam momentos em que o jogador possa precisar interromper as atividades de sua *thread* e esperar, associar isso com as primitivas ainda não parece muito adequado e no caso das associações do "Escalonando", elas poderiam estar ligadas a uma confusão com o termo preempção e a primitiva *sleep*.

Região crítica

Figura 44 – Associações feitas pelos alunos dos jogos com o tópico "Região crítica".



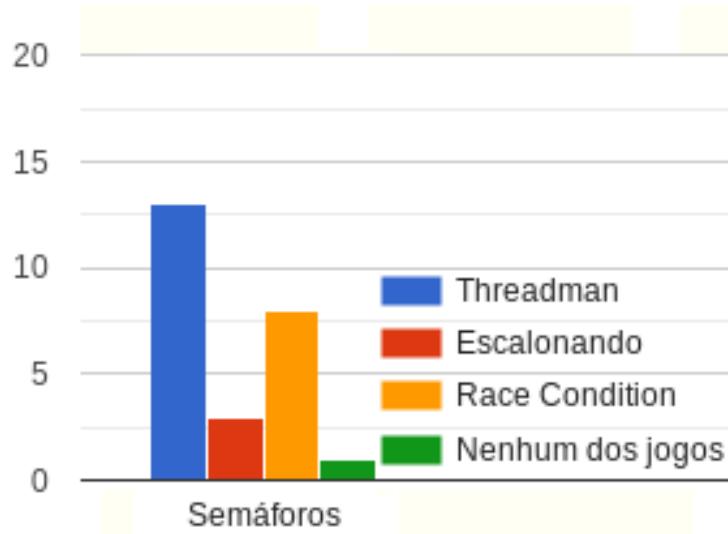
As associações esperadas seriam o "Threadman" e o "Race Condition", simultâneamente.

Foram feitas 18 associações ao "Race Condition", mas apenas 7 ao Threadman, com isso conclui-se que muitos alunos não assimilararam a ideia de que os semáforos estariam controlando o fluxo de *threads* em regiões críticas (conforme ilustrado na Figura 44).

Houve também 4 associações ao "Escalonando", talvez por conta da regra que apenas um processo pode ser executado por vez.

Semáforos

Figura 45 – Associações feitas pelos alunos dos jogos com o tópico "Semáforos".

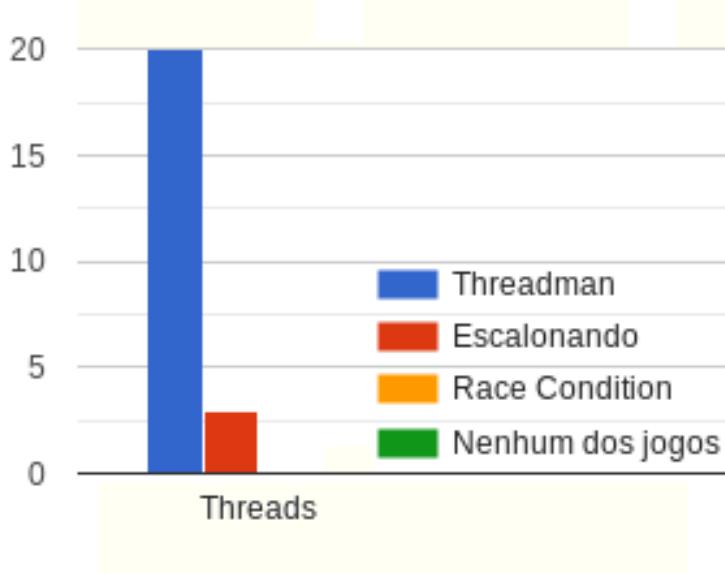


A associação esperada seria o "Threadman" e 13 alunos fizeram essa associação (conforme ilustrado na Figura 45).

Houve 3 associações ao "Escalonando" e 8 ao "Race Condition", uma possível explicação para as associações do "Race Condition" se dê pelo modo *strict alternation*, as cores do círculo que representam a região crítica variar. A mesma justificativa não cabe as associações ao "Escalonando", pois as cores nesse jogo não representam as permissões de uma região crítica, mas sim os estados de um processo.

Threads

Figura 46 – Associações feitas pelos alunos dos jogos com o tópico "Threads".



Os três jogos poderiam ser associados, mas o mais esperado certamente é a associação ao "Theadman". Dessa vez todos os 20 alunos realizaram essa associação e teve mais 3 associações ao "Escalonando", não houve outras associações (conforme ilustrado na Figura 46).

Variáveis de travamento

A associação esperada era ao "Race Condition" e o nome de um dos modos desse jogo é "Variáveis Lock". Um total de 13 alunos fez essa associação (conforme ilustrado na Figura 47).

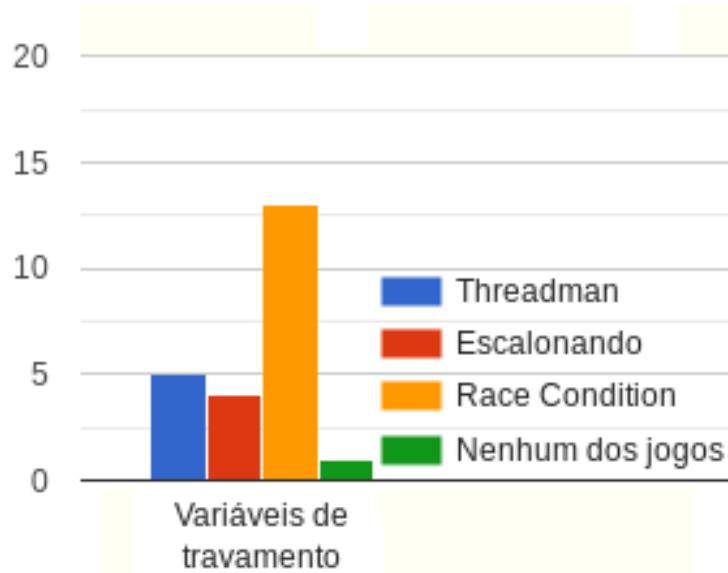
Houve, porém, 5 associações ao "Threadman", 4 ao "Escalonando" e 1 a "Nenhum dos jogos".

3.10.3 Perguntas dissertativas

Na terceira parte do formulário foram apresentadas questões dissertativas que tecnicamente poderiam ser respondidas caso o aluno interagisse com os jogos com a disposição de interligar os seus conhecimentos sobre a disciplina de Sistemas Operacionais com a mecânica dos jogos.

Nesse sentido, para responder essas questões, espera-se que o aluno observe com maior atenção o funcionamento desses jogos e jogue algumas partidas mudando a sua estratégia e

Figura 47 – Associações feitas pelos alunos dos jogos com o tópico "Variáveis de travamento".



verificando as vantagens e desvantagens ao adaptar essas estratégias.

Essas questões, portanto, foram desenvolvidas supondo que os usuários dos jogos já possuíssem conhecimento prévio da disciplina e que reforçaria esse conhecimento.

Assim, com as questões apresentadas nessa terceira parte do formulário espera-se verificar a eficiência na utilização dos jogos como ferramentas para ensinar também por meio da mudança de estratégia e comportamento dos próprios usuários.

Questão 8: Explique como é possível simular o comportamento dos algoritmos *first fit*, *best fit* e *worst fit* no jogo *Threadman*.

A simulação desses algoritmos seriam feitas no momento de gerenciar a memória no jogo.

Nesse momento, o jogador tem a possibilidade de escolher em que espaço da fila de memórias ele desejava depositar a informação recolhida. Assim, se ele depositasse a informação na primeira que coubesse, estaria simulando o comportamento do *first fit*, se ele escolhesse depositar naquele que tivesse o menor espaço restante possível, estaria simulando o *best fit* e se ele decidisse depositar naquele que tivesse com mais espaço sobrando, estaria simulando o *worst fit*.

Das 20 respostas recebidas, ocorreram 17 acertos, mas destes um total de 9 alunos não explicaram como a simulação de cada algoritmo funcionaria, esses deram respostas mais vagas, escrevendo que para simular os algoritmos é preciso gerenciar a memória, no entanto não dando instruções sobre a maneira como a memória precisava ser gerenciada, o que era esperado.

Essas 9 respostas apenas podem serem consideradas corretas pela margem de interpretação que a questão permitiu da maneira que o enunciado foi escrito.

Questão 9: Quais as vantagens e desvantagens dessas estratégias dentro do jogo?

First fit encontraria mais rapidamente um espaço para armazenar dada a informação e dado o instante, mas não apresentaria uma estratégia para aproveitamento dos espaços disponíveis.

Best fit e *worst fit* permitiriam melhor aproveitamento dos espaços disponíveis, mas o *best fit* também representaria um desperdício de memória internamente aos blocos e o *worst fit* representariam um desperdício da memória externamente ao bloco.

11 alunos acertaram a resposta, no entanto 4 dessas respostas não explicou as vantagens e desvantagens para cada estratégia como era esperado, só podendo ser consideradas respostas corretas uma vez que a margem de interpretação que o enunciado acabou por propor é ampla.

Questão 10: Explique como é possível simular o comportamento dos algoritmos FCFS, SJF e SRNT no jogo Escalonando.

Para simular o FCFS, o aluno deveria colocar em execução sempre os primeiros processos que entrarem nas filas, ou seja, os que se encontram na região inferior, para simular o SJF, o aluno deveria colocar em execução sempre os menores processos disponíveis e para simular o SRNT, o aluno deveria considerar quanto tempo levaria para concluir um processo que está em execução e se esse for maior do que um dos processos em espera, preemptá-lo.

Um total de 16 alunos respondeu corretamente, no entanto, 7 alunos não explicaram como a simulação de cada algoritmo funcionaria, apenas responderam que deveriam mudar a estratégia na hora de decidir que processo executar, no entanto sem especificar como. A resposta desses só pode ser considerada correta pela margem de interpretação que o enunciado possui.

Houve um total de 4 respostas incorretas, mas 3 desses alunos conseguiram explicar o funcionamento da simulação de dois dos algoritmos, não conseguindo explicar apenas o do SRNT.

Questão 11: Quais as vantagens e desvantagens dessas estratégias dentro do jogo?

Essa é uma questão que poderia ser vista por diferentes pontos, pois tecnicamente nenhuma dessas estratégias trazem vantagens reais para o jogo, mas considerando uma restrição nas regras em que o aluno se colocasse na posição de ter que escolher entre esses algoritmos, então as vantagens e desvantagens seriam notados.

Utilizando o FCFS, o jogador não teria vantagens frente ao jogo, mas considerando um Sistema Operacional, esse algoritmo garantiria que todos os processos seriam atendidos. A sua desvantagem aparece quando há processos de longa duração na fila que, dentro das regras do algoritmos, deverão ser atendidos conforme a ordem de chegada.

Com o SJF, os processos menores seriam atendidos antes, o que diminuiria a fila de maneira eficiente, no entanto, esse método faz com que se acumulem os processos grandes.

O SRNT teria a mesma vantagem e desvantagem do SJF, no entanto ele é mais complicado de aplicar no jogo e também provoca um desperdício de tempo ao preemptar um processo.

Uma resposta válida seria dizer que não há vantagens de usar esses algoritmos no jogo, outra seria apontar apenas as desvantagens. Com essas considerações, 12 alunos conseguiram responder essa questão corretamente sendo que 9 explicaram corretamente as vantagens e desvantagens dos algoritmos, 3 alunos apontaram apenas as desvantagens e 2 comentaram e justificaram o porquê de não haver vantagens nesse jogo em específico.

Questão 12: No jogo *Race Condition*, qual a maior dificuldade do modo *sleep wakeup*? Como associar essa dificuldade com a desvantagem dessa estratégia?

A desvantagem desse modo seria manter processos bloqueados e desbloqueados quando não deveriam estar assim.

Foram 15 alunos a responder corretamente a essa questão sendo que apenas 1 deles não associou a dificuldade com a desvantagem dessa estratégia na prática.

Questão 13: No jogo *Race Condition*, qual a maior dificuldade do modo variável de trava? Como associar essa dificuldade com a desvantagem dessa estratégia?

Nesse modo, quando mais de um processo está tentando acessar a região crítica em um intervalo curto de tempo, pode ocorrer de não haver tempo de se restringir o acesso antes da entrada do segundo.

Foram 15 alunos a responder corretamente a essa questão e 3 dos que erraram comentaram da formação de uma fila de processos, no entanto sem citar o problema de entrar mais de um processo dessa fila antes de conseguir restringir.

Questão 14: No jogo *Race Condition*, qual a maior dificuldade do modo estrita alternância? Como associar essa dificuldade com a desvantagem dessa estratégia?

Existem duas dificuldades para apontar nesse jogo, a necessidade de ter que variar o valor da estrita alternância até apontar para o processo que está livre e a necessidade de atenção ao fazer a transição dos valores da variável para que não ocorra de dois processos invadirem a região crítica. Apenas o primeiro se relacionaria com a disciplina, sendo que essa estratégia seria eficiente em evitar a entrada de mais de um processo na região crítica.

Foram 14 alunos a responderem de forma correta a questão, entretanto, 4 deles apontaram apenas para o segundo defeito e não fizeram a relação com a disciplina.

Por meio desses questões, foi possível observar que boa parte dos alunos que jogaram o jogo e responderam as questões estavam mesmo preocupados em aprender a matéria e a buscar avaliar se o aprendizado havia ocorrido ou não.

3.10.4 Pesquisa de sugestões

Na parte final, a quarta e última parte de questões, colocadas como opcionais, foi permitido que os alunos dessem sugestões e fizessem críticas aos jogos.

Questão 15: Indique aqui os pontos positivos dos jogos dessa atividade.

Foram 2 alunos que consideraram o material como uma boa ferramenta para reforço da disciplina, 5 consideraram o método como didático, 3 alunos elogiaram a estética, principalmente o visual, 2 alunos apontaram como uma boa maneira de revisar a disciplina e um aluno considerou um método de tirar dúvidas.

De modo geral foram críticas positivas e que permitem avaliar que os jogos podem ser ferramentas úteis para o ensino de Sistemas Operacionais.

Questão 16: Indique aqui os pontos negativos dos jogos dessa atividade.

Foram 3 os alunos a fazerem reclamações com relação a estética dos jogos, e foi apontado principalmente a questão musical, sendo que 2 alunos reclamaram da mecânica. Teve 1 aluno que não considerou que os jogos possuían clareza, 1 que apontou que os tutoriais poderiam ser melhores, 1 que não acredita que os jogos sejam muito intuitivos e 1 que considerou que sem conhecimento prévio talvez não conseguisse jogar.

Pode-se inferir que modificações e melhorias nos jogos são sempre necessárias para que os mesmos continuem atrativos e possam ser utilizados por vários alunos.

Questão 17: Indique aqui sugestões a respeito do jogo.

Os alunos fizeram uma série de sugestões interessantes para serem consideradas em trabalhos futuros. Elas serão discutidas posteriormente nesse documento no próximo capítulo, pois serão consideradas trabalhos futuros.

Questão 18: Escreva a seguir quaisquer *bugs* ou mal funcionamento observados nos jogos.

Os alunos também apontaram para alguns problemas que tiveram ao executar os jogos. Esses *bugs* serão discutidos também no Capítulo 4.

3.11 Considerações finais

Neste capítulo foram apresentadas as justificativas de cada escolha adotada ao projeto, foi explicado como cada tópico selecionado da disciplina de Sistemas Operacionais I é representado nos jogos e analisou-se o resultado obtido ao apresentar os jogos a alunos que cursavam a disciplina na turma de Engenharia de Computação no primeiro semestre de 2019.

Dos resultados obtidos, da finalização dos jogos e da experiência obtida enquanto esse projeto era desenvolvido, extraiu-se diversas conclusões que serão tratadas no próximo capítulo.

As respostas obtidas nas questões 17 e 18 também são discutidas no próximo capítulo, uma vez que elas representam propostas a serem tratadas em trabalhos futuros.

Capítulo 4

CONCLUSÃO

4.1 Contribuições

Os alunos que realizaram o teste e preencheram o formulário, apontaram para diversos pontos positivos do uso de jogos para ensino de questões importantes na disciplina de Sistemas Operacionais I. Nas respostas entregues, elogiaram os jogos dizendo ser uma boa ferramenta para reforço, revisão, visualização e para tirar dúvidas.

Nas pesquisas foi obtida uma boa margem de acertos, o que indica clareza na correlação da disciplina e das mecânicas dos jogos.

Ocorreu uma boa aceitação entre os alunos que realizaram a disciplina, considerando as notas 3, 4 e 5 na primeira parte do formulário, 90% dos alunos consideraram os jogos divertidos, 90% consideraram intuitivos, todos eles consideraram o conteúdo comprehensível, 95% afirmaram que os jogos lhes foram úteis ao aprendizado pessoal, também 95% indicariam esses jogos para os próximos alunos que realizarem essa disciplina. Com isso, é plausível concluir que os jogos completaram os seus objetivos especificados nesse documento.

Como contribuição principal tem-se o desenvolvimento e disponibilização de três jogos completos que abordam conteúdos essenciais da disciplina de Sistemas Operacionais I que podem ser utilizados por docentes e alunos, seja para o ensino, ou para o reforço dos conteúdos.

4.2 Relacionamento entre o Curso e o Projeto

Diversas das disciplinas presentes no curso de Engenharia de Computação da Universidade de São Paulo (USP), campus de São Carlos foram essenciais para o desenvolvimento desse projeto.

Primeiramente e mais claramente, os conteúdos da disciplina Sistemas Operacionais I foram estudados e revistos para que fossem reinterpretados nos jogos, como a própria proposta deste projeto já especificou.

O projeto também serviu como ferramenta para ensino dessa disciplina, ao ser disponibilizada para os alunos que a estavam cursando o primeiro semestre de 2019, turma da Engenharia de Computação.

Conceitos de Introdução a Ciência de Computação, Estruturas de Dados e Programação Orientada a Objetos foram cruciais no desenvolvimento dos códigos fontes, além disso, o conhecimento obtido ao longo do curso em técnicas de programação e código c, c++ e c# foram imprescindíveis para que este projeto fosse concluído.

4.3 Considerações sobre o Curso de Graduação

As disciplinas ministradas nesse curso e os docentes inspiraram a realização desse projeto. A origem da proposta veio de uma atividade dada pela professora Kalinka Regina Lucas Jaquie Castelo Branco, na disciplina de Sistemas Operacionais I para a turma de Bacharelado em Sistemas de Informação no segundo semestre de 2018. Essa atividade funcionou como projeto final da disciplina e, posteriormente, inspiração para esse projeto, o qual a professora Kalinka aceitou orientar.

O professor Júlio Cezar Estrela que lecionou a mesma disciplina para a turma da Engenharia de Computação no semestre seguinte, primeiro de 2019, concordou em possibilitar que os jogos e o formulário da pesquisa fossem disponibilizados aos seus alunos, além de auxiliar e dar sugestões para os jogos e a apresentação deles aos alunos, disponibilizando de seu tempo em várias oportunidades.

Os alunos que participaram da pesquisa propuseram ótimas observações e sugestões, além de apontar defeitos e *bugs*, o que será de ótimo auxílio para manutenção do projeto entregue. Ao que demonstra as respostas escritas, houve seriedade, respeito e dedicação na utilização dos jogos e na resposta ao questionário, o que favoreceu a formação das conclusões relatadas nesse documento.

Com isso, o ambiente universitário se mostrou favorável para o desenvolvimento e conclusão desse tipo de trabalho.

4.4 Trabalhos futuros

Com o ambiente se mantendo favorável, haverá muito o que se fazer em seguida tanto para a manutenção e disponibilização desses jogos, quanto para o possível desenvolvimento de outros, esses que inclusive poderiam abordar outras disciplinas desse curso.

As conclusões aqui registradas e a experiência adquirida poderá orientar novos projetos nesse escopo e guiar novas possibilidades ao que diz respeito à aprendizagem baseada em jogos para o ensino superior.

Como registrado anteriormente nesse documento, foi pedido no formulário para que os alunos registrassem sugestões para os jogos desse projeto. Foram obtidas 10 respostas, algumas envolvendo mudanças interessantes na jogabilidade, outras na estética geral.

Um dos alunos afirmou ser daltônico e comentou sobre a necessidade de trabalhar na palheta de cores, pois os três jogos usam bastante o verde e o vermelho. Outro aluno propôs a possibilidade de bonificar o jogador por usar estratégias baseadas em algoritmos reais. Houve diversas outras propostas a serem estudadas e que poderão ser aplicadas em novas versões desses jogos.

Também foi pedido para que os alunos apontassem *bugs* que lhes tivessem ocorridos durante a execução dos jogos. Foram apontados os seguintes: impossibilidade de fechar o tutorial no meio, música duplicada, travamento temporário na troca dos jogos, demora para o aparecimento de blocos no início do jogo "Escalonando", problemas em alguns blocos de memória do jogo "Threadman".

Consertar esses *bugs* certamente seria prioridade dos trabalhos futuros, bem como reforçar o que foi apontado como ponto positivo, trabalhar em cima do que foi dado como ponto negativo, rever sugestões, possibilitar o acesso e a disponibilização desses jogos e estudar a possibilidade de outras alternativas para a expansão do aprendizado baseado em jogos para o ensino superior.

REFERÊNCIAS

- BASBAUM, S. R. **Sinestesia, arte e tecnologia : fundamentos da cromossomia.** [S.l.], 2002. Citado na página 13.
- BORIN, J. **Jogos e resolução de problemas: uma estratégia para as aulas de matemática.** [S.l.], 2007. Citado na página 14.
- DIAS FELIPE DE ALMEIDA FREITAS, J. C. T. X. J. M. E. M. R. J. R. G. S. Caroline Maria Galiza de A. **A trilha sonora dos games: uma análise da influência e importância das trilhas sonoras no sucesso dos videogames.** [S.l.], 2014. Citado na página 37.
- DIX JANET FINLAY, G. D. A. R. B. A. **Human-Computer Interaction ed.3.** [S.l.], 2004. Citado na página 38.
- EDWARDS, S. H. **Using Software Testing to Move Students from Trial-and-Error to Reflection-in-Action.** [S.l.], 2004. Citado na página 13.
- GIMP. 2019. Disponível em: <<https://www.gimp.org/>>. Acesso em: 27/05/2019. Citado na página 18.
- GOOGLE. 2019. Disponível em: <<https://www.google.com/forms/about/>>. Acesso em: 27/05/2019. Citado na página 20.
- LAHTINEN KIRSTI ALA-MUTKA, H.-M. J. E. **A Study of the Difficulties of Novice Programmers.** [S.l.], 2005. Citado 2 vezes nas páginas 13 e 38.
- LUZ, A. R. da. **Linguagens Gráficas em Videogame: Nascimento, desenvolvimento e consolidação do videogame como expressão gráfica.** [S.l.], 2009. Citado na página 38.
- MATTAR, J. **Games em educação: como os nativos digitais aprendem.** [S.l.], 2010. Citado na página 24.
- MOREIRA, M. A. **Teorias de aprendizagem.** [S.l.], 1999. Citado na página 13.
- NETWORKREDUX. 2019. Disponível em: <<https://lmms.io/>>. Acesso em: 27/05/2019. Citado na página 18.
- PRENSKY, M. **Aprendizagem baseada em jogos digitais.** [S.l.], 2012. Citado 3 vezes nas páginas 14, 23 e 24.
- SCHMIDT, G. **The Future Is Bright for the Video Games of Yesterday.** 2018. Disponível em: <<https://www.nytimes.com/2018/12/02/business/retro-video-games.html>>. Acesso em: 14/05/2019. Citado na página 13.
- SENA SARAH SCHMITHAUSEN SCHMIEGELOW, G. M. B. C. d. P. R. P. L. d. S. F. A. P. F. Samara de. **Aprendizagem baseada em jogos digitais: a contribuição dos jogos epistêmicos na geração de novos conhecimentos.** [S.l.], 2016. Citado na página 14.

TANENBAUM, A. S. **Sistemas Operacionais Modernos ed.3.** [S.l.], 2010. Citado 5 vezes nas páginas [16](#), [17](#), [29](#), [51](#) e [53](#).

TANG MARTIN HANNEGHAN, A. E. R. S. **Introduction to Games-Based Learning.** [S.l.], 2009. Citado na página [13](#).

UNITY. 2019. Disponível em: <<https://unity.com/>>. Acesso em: 27/05/2019. Citado na página [18](#).

WHALEN, Z. N. **Play along: video game music as metaphor and metonymy.** 2004. Disponível em: <http://etd.fcla.edu/UF/UFE0004911/whalen_z.pdf>. Acesso em: 16/04/2019. Citado na página [37](#).