

# **Relatório Técnico**

## **Jogo das Somas 2.0**

**Naylane do Nascimento Ribeiro**

UEFS – Universidade Estadual de Feira de Santana  
Av. Transnordestina, s/n, Novo Horizonte  
Feira de Santana – BA, Brazil – 44036-900

naylaneribeiro4@gmail.com

***Resumo.** Este relatório técnico tem a finalidade de apresentar de forma geral como foi o processo de discussão, desenvolvimento e solidificação do Jogo das Somas 2.0, idealizado pelos irmãos Lara e Liam. Com ênfase na análise do problema, resultado dos debates em grupo, além da construção e funcionamento do sistema.*

### **1. Introdução**

Durante a pandemia do Covid-19, foi recomendado que a população entrasse em isolamento social, fazendo com que as pessoas buscassem formas de entretenimento dentro de casa. Os jogos se destacaram nesse período, visto que, segundo dados da Pesquisa Game Brasil [2022], durante a pandemia, 72,2% dos brasileiros afirmaram terem aumentado o consumo de jogos.

Independente do Covid-19, existem pessoas que permanecem dentro de casa durante longos períodos de tempo por conta de invernos rigorosos, como por exemplo os dois irmãos Lara e Liam. Devido ao tempo que Lara e Liam passam brincando dentro de casa, eles inventaram um novo jogo inspirado no Sudoku e no Jogo das Somas Esquecidas. Contudo, os irmãos não sabem programar, então foram convocados os alunos de Engenharia de Computação da Universidade Estadual de Feira de Santana (UEFS) para o desenvolvimento do programa.

Este relatório descreve o processo de desenvolvimento do Jogo das Somas 2.0 na linguagem de programação Python, na qual foi feita manipulação de matrizes e modularização. Este relatório está organizado nas seções: metodologia, resultados e discussões, além da conclusão.

### **2. Metodologia**

Para o desenvolvimento do jogo, foi utilizada a metodologia de ensino Problem Based Learning (PBL), onde foram realizadas sessões tutoriais com o intuito de discutir a melhor maneira de desenvolver o jogo. Inicialmente, foram discutidas as maneiras de criar matrizes, tendo a opção de fazer com lista de listas ou tuplas em dicionários. Após isso, as discussões giraram em torno de como obter a soma das linhas e colunas, sendo viável ter outro tabuleiro organizado de modo que permita-se automatizar o processo. Foi decidido em conjunto tentar implementar a forma otimizada de calcular as somas, porém também foi discutida acerca da maneira manual. Ademais, foi abordado o meio de revelar um número na seção escolhida pelo jogador, o qual foi consentido em unanimidade o uso de dois tabuleiros: um para armazenar os números e outro para usar

durante as partidas. O nome dado para os tabuleiros citados variam, contudo isso não atrapalhou o andamento das sessões já que o raciocínio é o mesmo.

Quanto às funcionalidades, o jogo é disputado entre dois jogadores: Jogador 1 e Jogador 2, utilizando um único tabuleiro, com ambos jogando no mesmo. O programa deve permitir ao usuário escolher qual o nível de dificuldade desejado. No nível 1 o tabuleiro possui as dimensões 4x4, dividido em 4 seções, e no nível 2 o tabuleiro apresenta as dimensões 9x9, dividido em 9 seções. Cada seção do tabuleiro deve conter números não repetidos ordenados de forma aleatória, sendo números de 1 a 4 para o tabuleiro de 4x4, e números de 1 a 9 para o tabuleiro de 9x9. Respectivamente, ao lado de cada linha e abaixo de cada coluna do tabuleiro é exibido os valores das suas somas. A cada rodada um jogador escolhe uma seção e o número a ser revelado na seção escolhida. Se o número revelado completar uma linha e/ou coluna do tabuleiro, o valor da soma da linha e/ou coluna é acumulado na pontuação do jogador da rodada. Além disso, se o número revelado completar a diagonal principal do tabuleiro, o jogador pontua o dobro do valor da soma da diagonal. Vence o jogador que tiver a maior pontuação após o tabuleiro ser completamente revelado.

Para o cumprimento dessas funcionalidades, elas foram divididas em partes e cada uma foi trabalhada por vez. Inicialmente, foram implementadas as funções responsáveis pela execução da partida do jogo, como a criação e a impressão do tabuleiro, gerador dos números que preenchem o tabuleiro e o cálculo das somas das linhas, das colunas e da diagonal. Após isso, foi implementada a função que executa a partida, onde foram feitas chamadas às funções criadas anteriormente, e o menu principal do jogo. Por fim, o programa foi devidamente testado e os problemas encontrados foram resolvidos quando possível.

Para a implementação do programa foi utilizado o editor de código-fonte Visual Studio Code no sistema operacional Windows.

### **3. Resultados e Discussões**

Pensando em fazer a partida iniciar com o tabuleiro tendo os valores escondidos e revelar os mesmos de acordo com o andamento das rodadas, foi feita a implementação de dois tabuleiros: o tabuleiro espelho e o tabuleiro jogável. Após ser gerado, no início de uma partida, o tabuleiro espelho é imutável e guarda os números randomizados, enquanto o tabuleiro jogável é mutável e mostrado para os jogadores durante toda a partida. Assim, após o jogador informar o número que deseja revelar em uma determinada seção, as seções são percorridas e, quando encontrada a posição do número no tabuleiro espelho, o tabuleiro jogável tem seu valor alterado nessa mesma posição, ou seja, o número é revelado. Os ditos tabuleiros foram gerados com uma lista de listas para construir a matriz, onde a lista maior se refere ao tabuleiro e cada lista dentro da mesma corresponde às seções.

Para garantir que os números não se repetissem na mesma seção os mesmos são gerados com um loop for utilizando a função range do Python, onde a sequência de números começa de 1 e vai até a dimensão do tabuleiro (4 ou 9). Foi importada a função shuffle da biblioteca Random para ordenar os ditos números aleatoriamente.

Referente às somas das linhas e colunas, seus valores são armazenados em listas, que inicialmente são preenchidas por zeros e depois pelo valor da soma correlato. O cálculo das somas é feito dentro do comando for, usando a função range, onde, a cada iteração do loop, é feita a soma de uma linha e coluna manipulando os índices do tabuleiro espelho. Um elemento de uma matriz é conseguido pelos índices da mesma e, no caso do tabuleiro, o primeiro índice se refere a seção, enquanto o segundo corresponde a posição dentro da seção e é alterado de acordo com a variável do laço.

A posição das linhas do tabuleiro 4x4 é obtida por uma “fórmula” aplicada no segundo índice do tabuleiro espelho, multiplicando a variável do laço por 2 e somando 1 quando necessário “pular” para a posição ao lado, percorrendo assim a matriz horizontalmente. É possível visualizar na prática como ficam os índices do tabuleiro espelho na figura 1, na qual as partes em vermelho são a “fórmula”. O cálculo das somas das linhas no tabuleiro 9x9 segue a mesma lógica.

$[0][x*2]$ $[0][0*2]$	$[0][(x*2)+1]$ $[0][(0*2)+1]$	$[1][x*2]$ $[1][0*2]$	$[1][(x*2)+1]$ $[1][(0*2)+1]$
$[0][1*2]$	$[0][(1*2)+1]$	$[1][1*2]$	$[1][(1*2)+1]$
$[2][x*2]$ $[2][0*2]$	$[2][(x*2)+1]$ $[2][(0*2)+1]$	$[3][x*2]$ $[3][0*2]$	$[3][(x*2)+1]$ $[3][(0*2)+1]$
$[2][1*2]$	$[2][(1*2)+1]$	$[3][1*2]$	$[3][(1*2)+1]$

**Figura 1. Posição das linhas do tabuleiro 4x4**

Em relação a posição das colunas do tabuleiro 4x4, também foi chegada em uma “fórmula”, onde o segundo índice é dado pela variável do laço e, quando necessário pular para a posição abaixo, se soma 2, formando assim um loop que percorre a matriz verticalmente. É possível visualizar na prática como ficam os índices do tabuleiro espelho na figura 2. O cálculo das somas das colunas no tabuleiro 9x9 segue a mesma lógica.

[0][x] [0][0]	[0][1]	[1][x] [1][0]	[1][1]
[0][x+2] [0][0+2]	[0][1+2]	[1][x+2] [1][0+2]	[1][1+2]
[2][x] [2][0]	[2][1]	[3][x] [3][0]	[3][1]
[2][x+2] [2][0+2]	[2][1+2]	[3][x+2] [3][0+2]	[3][1+2]

**Figura 2. Posição das colunas do tabuleiro 4x4**

Para saber quando uma linha, coluna e/ou diagonal foi revelada, a cada rodada é chamada uma função que soma os atuais valores do tabuleiro jogável. A diferença entre a função que realiza o somatório das linhas, colunas e da diagonal para a esta que retorna os pontos está no tabuleiro utilizado. As somas são calculadas somente uma vez e são imutáveis durante a partida, já os pontos são calculados de acordo com o tabuleiro mutável a cada rodada. Esse método só é possível porque o tabuleiro jogável é inicialmente preenchido por zeros.

As linhas e colunas reveladas são guardadas em um dicionário. Assim, quando a soma e o ponto correspondente tiverem valores iguais, é verificado se já existe aquela linha/coluna no dicionários das reveladas, se não, o jogador da partida recebe a pontuação equivalente e a linha/coluna é adicionada no dicionário das reveladas. A pontuação da diagonal se dá da mesma forma, porém usando um boolean ao invés de dicionário.

Para jogar corretamente, devem ser digitados valores inteiros, para a seção e o número que deseja revelar, entre 1 e a dimensão do tabuleiro equivalente ao nível que está sendo jogado. Para tal, é implementada uma função responsável por validar as entradas do usuário.

Apesar de algumas tentativas, não foi desenvolvida a otimização da impressão do tabuleiro, sendo o mesmo impresso de forma manual pelos seus índices.

#### 4. Conclusão

Levando em conta os requisitos e funcionalidades solicitados, interpreta-se que o sistema cumpre com seus objetivos, mesmo que de forma não tão otimizada. Pensando em uma possível melhoria do código, seria interessante automatizar a impressão dos tabuleiros e otimizar mais o cálculo das somas das linhas e colunas. Além do mais, adicionar cores e a opção de digitar um nome para os jogadores tornaria o jogo mais atrativo.

## **Referências**

Pesquisa Game Brasil (2022) “Report PGB 2022”,  
<https://www.pesquisagamebrasil.com.br/pt/>, Setembro.