



ESTA013-17 - Fundamentos de Robótica

**Projeto - Jogo de Campo Minado com Braço Robótico**

Turma B - Matutino  
Prof. Dr. Sandro Luís Vatanabe

Luana Larissa Mendonça Teixeira 11201811394  
Alex Key Araki 11019914

## 1. Objetivos

O projeto consiste na programação do jogo conhecido como campo minado, um jogo de “sorte” no qual o jogador interage com um tabuleiro matricial, escolhendo uma casa de cada vez, podendo esta conter uma bomba ou não, caso encontre uma bomba, o jogo acaba, caso não, poderá ou não dar dicas, dependendo da posição das bombas, até que o jogo termine.

Ao implementar o jogo ao braço robótico, o modo como o jogador interage com a interface será diferente, uma vez que ao invés do jogador utilizar de uma interface digital em uma tela de computador, o braço utilizará de uma caneta e um quadro, onde será desenhado o tabuleiro do jogo e marcada as casas escolhidas.

A escolha pelo jogo do campo minado tem como objetivo principal aplicar os conhecimentos adquiridos ao longo da disciplina, assim como familiarizar com o manuseio e programação de robôs, neste caso, um ABB IRB1600 (6 kg e 1,45m).

## 2. Descrição do Projeto

O jogo foi dividido em vários grupos diferentes para manter a organização do código, sendo eles: `DesenhaTabuleiro`, `DesenhaDica`, `DesenhaBomba`, `DesenhaSafe`, `AplicarBomba`, `VerificarBomba` e o `main`.

Temos como principal o `main`, onde é chamado todos os outros e faz o jogo funcionar, inicia o jogo, desenha o tabuleiro, insere as bombas e dicas, e entra em um laço de interação com o jogador, aonde este insere coordenadas e o jogo confere se há bombas, dicas, ou não há nada na casa escolhida.

Há 4 grupos apenas para o braço desenha, sendo o primeiro o desenho do tabuleiro, denominado de `DesenhaTabuleiro`, o código nele é dinâmico, sendo possível desenha tabuleiros de qualquer tamanho com casas de tamanho variável, sendo necessário apenas alterar as variáveis nele inseridas.

Para desenha as dicas há 3 possibilidades em `DesenhaDica` separadas por condicionais, uma para casou houver a dica de 1 bomba, para 2 e 3. Seus tamanhos são fixos, tendo apenas como variáveis as posições onde elas serão desenhadas, valores acima não foram feitos devido ao fato de serem casos raros.

Em `DesenhaBomba` e `DesenhaSafe` os códigos são semelhantes ao citado acima, porém com apenas uma possibilidade de desenho para cada, no primeiro o braço desenhara uma bomba caso a posição da matriz escolhida seja 9, e um V para caso o valor seja 0, suas variáveis também referem-se à posição onde serão desenhadas.

Há também como o jogador inserir jogos da forma que quiser em `AplicarBomba`, nele estão contidos as matrizes de jogos prontos para tabuleiros 5x5 e 9x9, um detalhe importante é que o tabuleiro montado na matriz é o inverso do jogo real na máquina, uma vez que para o jogo ficar de frente para o jogador fora do alcance dos braços da máquina para evitar acidentes, o jogo foi escrito de maneira inversa ao eixo Y e X da máquina.

Por fim há o VerificarBomba, nele o código verifica qual a entrada fornecida pelo jogador e aplica ele a matriz, por ele que o jogo define se desenhará uma dica, uma bomba ou um V na coordenada escolhida. Nele também é feita a contagem de pontos, uma vez que todas as casas livres sejam escolhidas, o jogo termina parabenizando o jogador, caso o jogador escolha uma casa com bomba, o jogo também finaliza, porém dando Game Over.

O main contém também a possibilidade de o jogador pular a etapa de desenhar o tabuleiro e ir direto para o jogo, para caso o jogador perder e quiser apenas apagar os desenhos nas casas ao invés de apagar o tabuleiro todo, já que para desenhar tudo de novo leva um tempo considerável.

No main também há uma proteção contra o jogador inserir valores além do permitido para as matrizes, o que acarretaria em erro no jogo, por meio de laços e condicionais, o jogo repetirá o pedido de coordenadas até que o jogador insira valores permitidos.

O jogo se assemelha em quase tudo ao jogo Campo Minado do Windows na jogabilidade, faltando apenas a aplicação de quando jogador escolhe uma casa sem bombas perto, revelar todas as dicas próximas.

### 3. Fluxograma de Programação

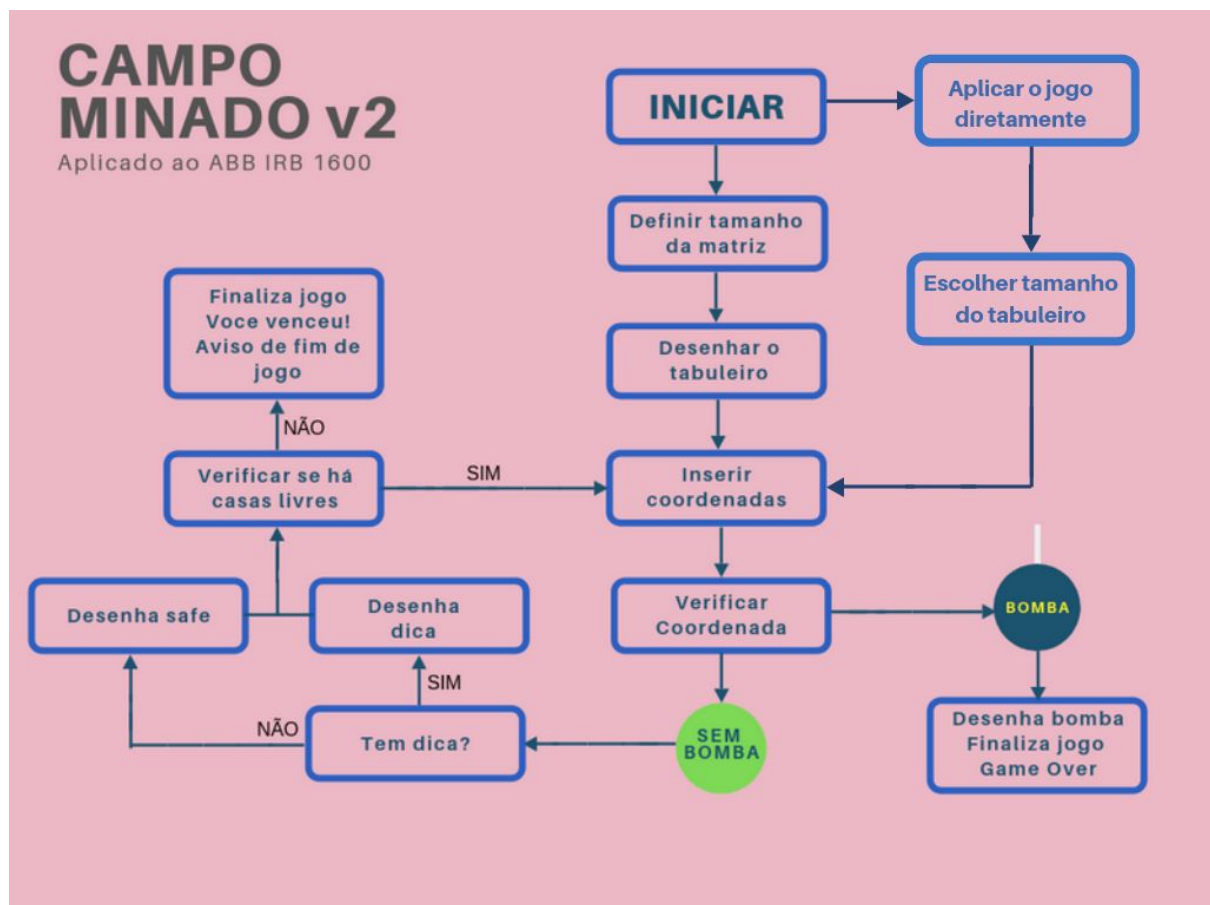


Figura 1 - Fluxograma do jogo Campo Minado na primeira versão.

Figura 5 - Fluxograma da versão final do jogo.

#### 4. Desafios Encontrados

- Aplicação de matrizes na criação de tabuleiros.
- Relacionar posições das coordenadas a posições na matriz.

Inicialmente o jogo estava escrito utilizando um array ao invés de matriz, pois estávamos encontrando dificuldades de aplicar matriz nesta linguagem, o uso de um vetor fazia o código ficar consideravelmente maior, além de mais limitado em relação às possibilidades de uso. Com a ajuda do professor, o problema foi resolvido e aplicada a matriz, o jogo se tornou mais dinâmico e deu novas possibilidades, que também seriam executáveis em um vetor, porém seria mais trabalhoso e menos eficiente.

Com a aplicação de matrizes a forma como o jogo percorre as posições e aplica a verificação se tornou mais fácil e o código relativamente menor, uma vez que anteriormente havia um código razoavelmente grande para verificação e aplicação de bombas na matriz, além do que pela matriz ter semelhança visual ao tabuleiro, é muito mais intuitivo na hora de escrever e entender o código.

Outro desafio foi ao relacionar posições da matriz à posições no tabuleiro, uma vez que o ponto de origem da máquina não ser a intuitiva para nós, como também para a interface ficar de frente para o jogar, ela teria de ser a inversa da máquina, a forma mais simples de resolver o problema, foi utilizar as posições de forma inversa, ou seja, decrescente. Sendo assim o novo ponto de origem para o jogo tornou-se o canto inferior direito, crescendo de forma linear até o ponto superior esquerdo, tendo como referencial o jogador de frente para o braço.

25	24	23	22	21	y	5
20	19	18	17	16		4
15	14	13	12	11		3
10	9	8	7	6		2
5	4	3	2	1		1
x	5	4	3	2	1	0

Figura 2 - Esquema do tabuleiro 5x5.

81	80	79	78	77	76	75	74	73	9 y
72	71	70	69	68	67	66	65	64	8
63	62	61	60	59	58	57	56	55	7
54	53	52	51	50	49	48	47	46	6
45	44	43	42	41	40	39	38	37	5
36	35	34	33	32	31	30	29	28	4
27	26	25	24	23	22	21	20	19	3
18	17	16	15	14	13	12	11	10	2
9	8	7	6	5	4	3	2	1	1
x	9	8	7	6	5	4	3	2	1
									0

Figura 3 - Esquema do tabuleiro 9x9.

Também foram pensados várias outras implementações para se fazer no jogo que podem ser aplicadas por grupos futuros, como a implementação de aleatoriedade, e ao pisar em uma casa sem bombas próximas, o jogo automaticamente revelar todas as dicas próximas.

Uma forma pensada e até testada para a aleatoriedade foi a de implementar uma equação senoidal com amplitude do tamanho do número de casas, por exemplo, para uma mesa 9x9, uma equação com amplitude de 0 a 81 e pegando apenas o intervalo entre os picos, para evitar que sejam sorteados valores iguais. Apenas não foi de fato aplicado pois não foi feito um código de aplicação de dicas automático para cada distribuição aleatoria.

## 5. Vídeo de Apresentação

Segue link da apresentação do projeto:

<https://youtu.be/vRQfG5eJK4c>