### **UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

### FERNANDO ITIRO NISHIMURA

### **Projeto Estrutura de Dados**

Jogo “Snake”

### **CURITIBA – PR**

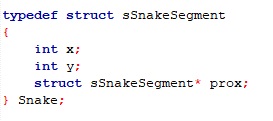
### **2021**

# INTRODUÇÃO

Neste documento se está relatado em como fora implementado o programa “Jogo Snake” como projeto final para a disciplina Estrutura de Dados e Algoritmos (ELP33), onde fora utilizado a estrutura Lista Simplesmente Encadeada como principal lógica e funcionamento do mesmo.

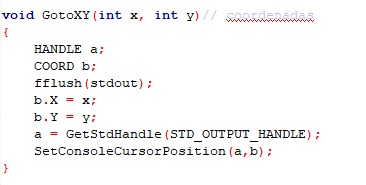
# DESCRIÇÃO E CONCEITOS

O “Jogo Snake” consiste em o usuário controlar a direção em que a cobrinha deve percorrer o tabuleiro deve seguir a fim de comer a comida. Em termos mais técnicos, as coordenadas ‘x’ e ‘y’ da cabeça da snake deve ser o mesmo que a da comida. O jogador deve repetir tais feitos a fim de ganhar pontos. A cada vez que a snake come a comida, seu corpo cresce, dificultando o movimento. Colisões com os limites do tabuleiro ou com o próprio corpo da cobra significa fim de jogo.



*Fig. 1: struct nó da Lista Encadeada. Corpo da cobra.*

O deslocamento do corpo, a imagem do tabuleiro, e identificação das coordenadas foram feitas a partir do auxilio da função gotoxy da <windows.h> e seus auxiliares.



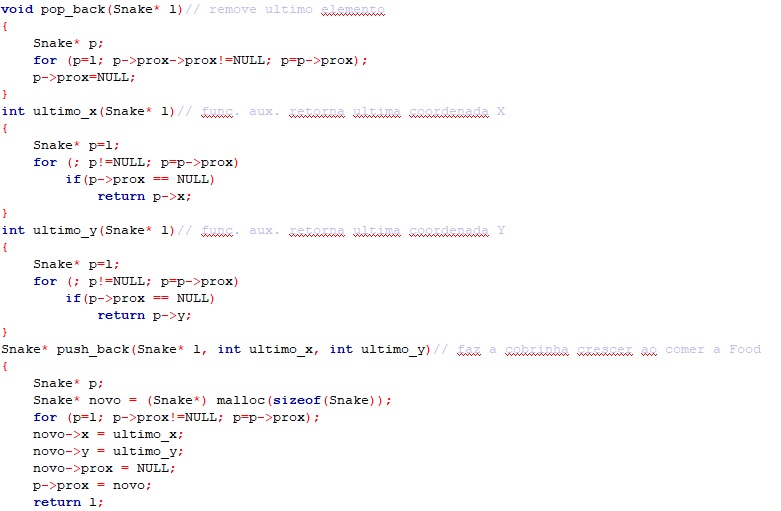
*Fig. 2: função GotoXY.*

Para a lógica principal do jogo, o movimento, deslocamento, a cabeça da snake fica no início da Lista Encadeada, o primeiro elemento. Assim, para dar a noção de deslocamento uma nova cabeça é inserida no início a cada frame, de acordo com a direção do input, e o ultimo pedaço de sua cauda é removida do final da lista. Tal lógica pode ser diferente do senso comum em que: ao comer a comida se é inserido mais um pedaço da cauda no início da lista, porém tal método se demonstrou muito complexo para quando se é necessário mudar de direção, já que para isso todos as coordenadas do corpo teriam que ter suas informações atualizadas por frame.

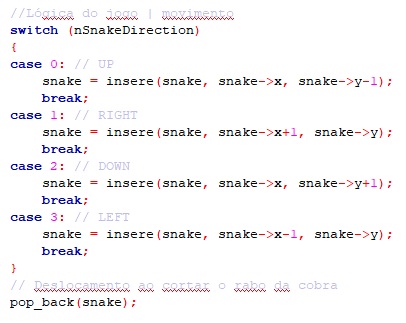
O input das direções em tempo real, direita e esquerda é feito via função KeyAsyncKeyState.



*Fig. 3: leitura do input pelo usuário.*

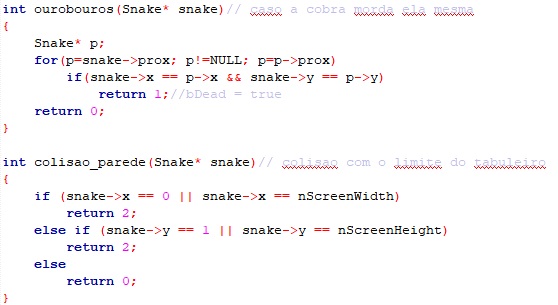


*Fig. 4: funções auxiliares ao deslocamento*

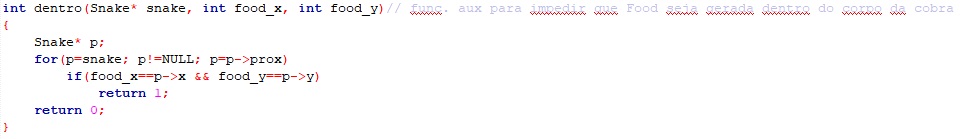


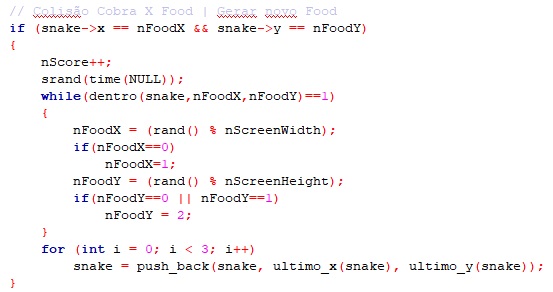
*Fig. 5: Lógica do movimento*

Para a checagem de fim de jogo: A cobra não deve encostar nos limites do tabuleiro e não deve morder o próprio corpo. Também se foi implementado o código de meio a comida não ser gerada dentro do corpo da snake, fora do tabuleiro, ou dentro do mesmo.



*Fig. 6: funções de checagem de fim de jogo.*



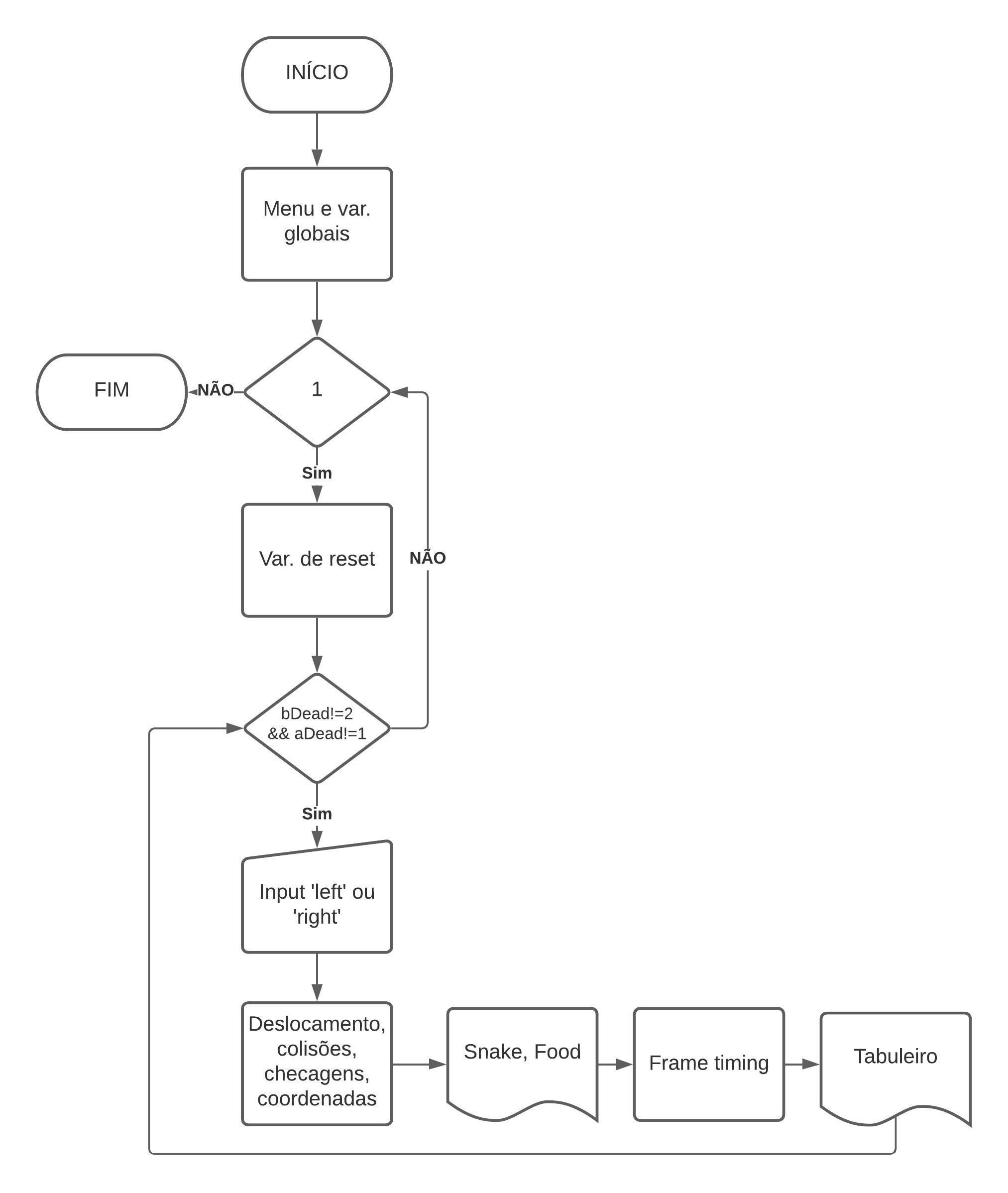


*Fig. 7: Prevenção de bug na geração da comida.*

# RESULTADOS

## 

*Fig. 8: Funcionamento do jogo.*



*Fig. 9: Fluxograma.*

# REFERENCIAS

JAVIDX9. “Code-It-Yourself! Snake! - Programming from Scratch (Quick and Simple C++)”. Youtube channel (javidx9). Disponível em: https://www.youtube.com/watch?v=e8lYLYlrGLg&t=891s