

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
INSTITUTO METRÓPOLE DIGITAL  
ESTRUTURAS DE DADOS BÁSICAS I  
LINGUAGEM DE PROGRAMAÇÃO I

**PROJETO SNAKE**  
**DOCUMENTAÇÃO DO CÓDIGO FONTE**

Joel Rodrigues de Lima Neto  
Francleide Peixoto Simão

NATAL – RN  
JUNHO/2015

## Índice

Enumeração Orientation .....	3
Estrutura Food .....	3
Atributos .....	3
Estrutura Node .....	3
Atributos .....	3
Classe Snake .....	3
Atributos .....	3
Métodos .....	3
Classe Map .....	3
Atributos .....	3
Métodos .....	4
Classe Game .....	4
Atributos .....	4
Métodos .....	4

## **Enumeração Orientation**

- Representa a orientação de cada bloco do corpo da cobra que pode ser Up, Right, Down ou Left.

## **Estrutura Food**

### **Atributos**

- int X e Y: Variáveis que armazenam a posição horizontal (X) e vertical (Y) do alimento da cobra.

## **Estrutura Node**

### **Atributos**

- int X e Y: Variáveis que armazenam a posição horizontal (X) e vertical (Y) de cada bloco que compõe o corpo da cobra.
- Orientation Direction: Variável que armazena a posição do nó do corpo da cobra.
- Node \*Next: Ponteiro que aponta para o próximo elemento do corpo da cobra.

## **Classe Snake**

### **Atributos**

- Node \*head: Ponteiro que aponta para a cabeça da cobra, que atua como guia do restante do corpo.
- Node \*tail: Ponteiro para a cauda da cobra, utilizado como referência para adicionar novos blocos ao corpo da cobra.
- int size: armazena o tamanho da cobra.

### **Métodos**

- Snake(): Construtor padrão da classe, onde é criada a cabeça (head) e adicionados 2 blocos ao corpo da cobra (tamanho inicial será 3).
- Node\* GetHead(): Método público para expor o ponteiro para a cabeça da cobra.
- void Eat(): Adiciona um novo bloco ao final do corpo da cobra (após a cauda).

## **Classe Map**

### **Atributos**

- int width: Largura do mapa.
- int height: Altura do mapa.
- int \*\*matrix: Matriz que armazena o conteúdo/estado de cada posição do mapa, onde o valor 0 indica uma posição vazia e 1 indica que há uma comida naquela posição.
- Food \*currentFood: Ponteiro para a comida que está atualmente aparecendo no mapa para ser alcançada pela cobra.

### **Métodos**

- `Map(int _width, int _height)`: Construtor sobrecarregado da classe, que recebe os valores da largura e altura, que são repassados para as variáveis `width` e `height` respectivamente.
- `int GetWidth()`: Método público para expôr a largura do mapa.
- `int GetHeight()`: Método público para expôr a altura do mapa.
- `Food GetCurrentFood()`: Método público para expôr a comida atual (`currentFood`).
- `void AddFood()`: Adiciona uma nova comida no mapa em uma posição aleatória.

### **Classe Game**

#### **Atributos**

- `int **matrix`: Matriz que armazenará todos os elementos do jogo, incluindo as posições do mapa, a comida e o corpo da cobra. Essa variável será autlizada durante o Game Loop e é a partir dela é que o jogo será desenhado.
- `Map *map`: Objeto local que indica o mapa do jogo atual.
- `Snake *snake`: Objeto local que indica a cobra do jogo atual.

#### **Métodos**

- `Game()`: Construtor padrão da classe, onde são iniciados os elementos do jogo.
- `Update()`: Método integrante do Game Loop e onde são atualizados os valores das variáveis.
- `Draw()`: Segundo método do Game Loop e é responsável por desenhar o jogo com base nas variáveis locais.