

	UNIVERSIDADE ESTADUAL DE SANTA CRUZ	
	Disciplina: Estrutura de Dados	
	Professora: Luenne Nailam	
	Discente:	Matrícula:
	Curso: Ciência da Computação	Semestre: 2025.2

## Projeto do Semestre – CodeBattle – A Conquista da Árvore Binária

### Objetivo Geral

Desenvolver um jogo interativo, em linguagem C, baseado em uma árvore binária de territórios, no qual dois jogadores competem para conquistar o território central (a raiz da árvore). O projeto visa integrar os principais conceitos abordados ao longo da disciplina: ponteiros, alocação dinâmica, tipos abstratos de dados, listas, pilhas, filas, recursividade e estruturas de árvores.

### Descrição Geral do Jogo

O jogo **CodeBattle** simula um mundo representado por uma árvore binária completa de altura 3 (com 15 nós), onde cada nó é um território com propriedades específicas. Dois jogadores começam em posições opostas (nós folha) e devem subir até o nó raiz (chamado de *Núcleo-X*), coletando recursos e enfrentando desafios no caminho.

### Estrutura da Árvore

A árvore binária será criada dinamicamente utilizando alocação de memória. Cada nó da árvore é um território com os seguintes atributos:

- Nome do território (string)
- Tipo (RECURSO, NORMAL, BATALHA)
- Ocupado (0 = vazio, 1 = jogador 1, 2 = jogador 2)
- Ponteiros para os filhos esquerdo e direito

A árvore possui 15 nós, organizados em 4 níveis (nível 0 a 3), sendo o nível 0 o da raiz. Jogadores iniciam nos nós folhas (nível 3): jogador 1 à esquerda e jogador 2 à direita.

### Mecânica do Jogo

#### Objetivo dos Jogadores

Cada jogador deve alcançar o território **Núcleo-X** antes do adversário. Para isso, deve percorrer a árvore, subir nível a nível, passando pelos nós intermediários, de acordo com regras de movimentação e estado dos territórios.

## Movimentação

- A cada turno, um jogador pode tentar subir um nível, indo para o território superior imediato.
- Como não há ponteiro para o pai, o jogador deve usar funções recursivas para localizar o nó-pai a partir da raiz.
- Não é possível passar por um território já ocupado pelo outro jogador.

## Tipos de Território

- **NORMAL**: passagem livre.
- **RECURSO**: o jogador coleta um item (arma, escudo, energia) e adiciona à sua lista de inventário (usar TAD Lista).
- **BATALHA**: se o território estiver protegido, o jogador deve verificar se possui recursos suficientes para conquistar o território (ex: escudo + arma). Caso contrário, volta ao nó anterior (usar TAD Pilha para armazenar histórico).

## Inventário e Histórico

- Os jogadores devem manter um inventário de recursos, usando uma estrutura de lista encadeada.
- A movimentação do jogador deve ser registrada usando uma pilha (voltar ao nó anterior, desfazer ações).

## Implementação Técnica

O código deve utilizar obrigatoriamente os seguintes conceitos:

- Alocação dinâmica de memória
- Ponteiros para estruturas
- Tipos abstratos de dados (listas, pilhas e filas)
- Estrutura de árvore binária com nós compostos
- Recursividade (principalmente para busca de caminhos e nós-pai)
- Modularização: separar funções em arquivos `.c` e `.h`

## Regras do Projeto em Sala

- O projeto será desenvolvido em grupos de até 3 estudantes.
- Serão destinados 4 períodos (2 encontros) para o desenvolvimento em sala de aula.
- A entrega deve conter o código-fonte, um resumo do funcionamento do jogo.
- Serão avaliados: domínio dos conceitos, organização do código, funcionalidade do jogo, e capacidade de trabalho em equipe.

## Critérios de Avaliação

- Aplicação dos conceitos da disciplina
- Funcionamento do jogo (execução sem erros)
- Estrutura e organização do código (modularização)
- Criatividade e documentação